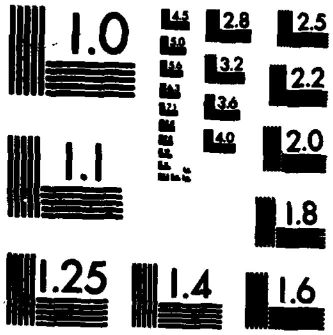
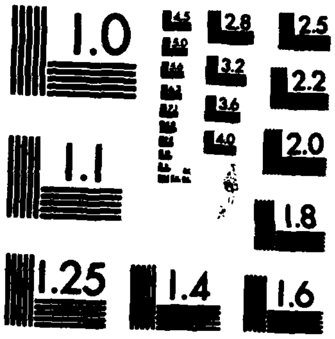
AD-A120 771   COMMUNICATIONS PROTOCOL STRUCTURE(U) CONTEL INFORMATION   1/2
SYSTEMS VIENNA VA 23 SEP 82 FR.343.01 SBI-AD-E001 358
N00014-81-C-2477

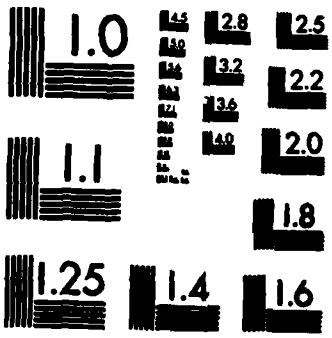
UNCLASSIFIED   F/G 17/2   NL



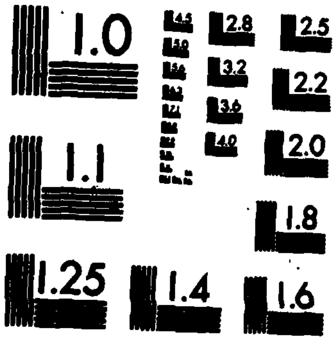
MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A



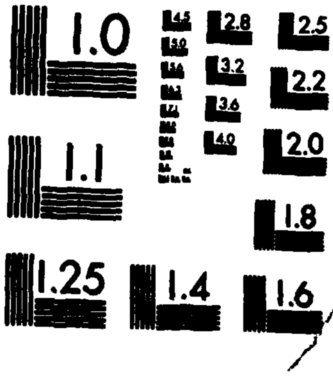
MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

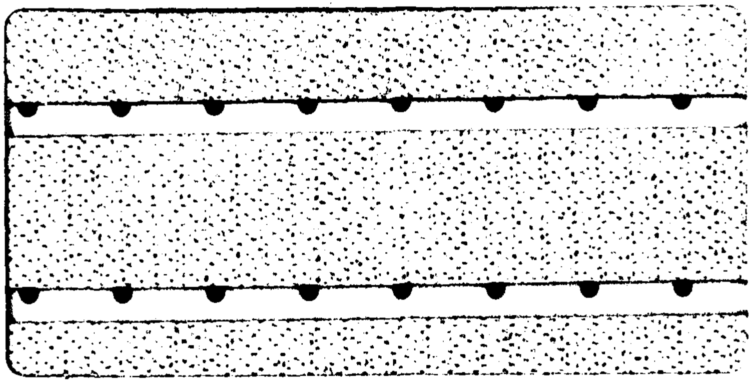


MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

E001358 ②



CONTEL
INFORMATION SYSTEMS

CONTEL
INFORMATION SYSTEMS

FINAL REPORT

COMMUNICATIONS PROTOCOL STRUCTURE

Performed for

Naval Research Laboratory
Washington, DC
Contracting Officer: Ms. Virginia Dean
Contract No. N00014-81-C-2477

FR.343.01
September 23, 1982

CONTEL INFORMATION SYSTEMS
301 Tower Building
Vienna, VA 22180

CONTEL
INFORMATION SYSTEMS

EXECUTIVE SUMMARY

CONTEL
INFORMATION SYSTEMS

EXECUTIVE SUMMARY

This is the final report summarizing the analysis and results of the four tasks performed under contract number NOOO14-81-C-2477. The report is composed of three volumes. The first volume outlines the requirements and protocol structure for the Naval Telecommunications System (NTS). The second volume outlines the requirements and the specifications of the high frequency (HF) ship-shore protocol. The third volume examines the issues related with the integration of voice/data.

The applicability of the Open Systems Interconnection (OSI) protocol architecture developed by the International Standards Organization has been studied for potential use in future Naval Telecommunications Systems. The major conclusions are:

- The overall protocol architecture is sufficiently general to accommodate NTS requirements.

- The lower level protocols will typically be more complex than required in commercial networks; in particular, we propose an enhanced physical layer.

- The enhanced physical layer should not be broken into additional sublayers for standardization because doing so would impose unnecessary constraints on the systems implementer.

- The network and transport layers should both be broken out into intra-network (and inter-network layers).

- Accountability functions must be built into higher layers.

- All platforms will require up to the applications layer, including satellites which require it for monitor and control information processing.

- The use of passive acknowledgements of the link level inherently violates the OSI architecture; the resolution of this issue is open.

Volume II contains requirements, environment and issues related to the HF ship-shore protocol. Various alternative protocols were evaluated. The fundamental design decisions that guided the development of the eventual HF ship-shore protocol consist of:

- Devising a protocol that combines the robustness of polling in the presence of a high transmission error rate and the fairness (in a global allocation of capacity) of Demand Assignment Multiple Access schemes.

- Capability of transparently handling all data offered for transmission.

- A selective repeat link control procedure for efficiency in the high error HF environment.

- In particular, a unique acknowledgement protocol has been developed to enable the sender to quickly detect when a transmission error occurs, and subsequently retransmit the frame. This improves performance because frames must be delivered to the higher level protocol in order; hence, to minimize message delay, such errors should be corrected quickly.

- The prioritization of messages is effected at the network level because of the complexities involved in handling different priorities in the same link and also delivering frames in order; an alternative solution is to employ separate links for different priority traffic.

- A unique expanding slot net entry protocol has been developed to provide prompt entry to the network with an efficient use of channel capacity.

- Incorporation of fields in the frame header for flow control although specifications for setting such fields will be dependent upon the buffer management scheme to be developed; this latter issue is deemed to be an implementation issue.

The major issue yet to be resolved is the quantitative performance of the protocol, especially the rules for allocating capacity. This is an especially interesting problem because the shore may be transmitting (and receiving) at different code rates during the polling cycle. In particular, the robustness of the scheme in terms of the ratio of ship to shore traffic relative to shore to ship traffic and establishment of an optimal code rate should be studied. The recommended approach is to simulate the protocol over a wide range of traffic loads. The second volume contains a description of the various components of the HF ship-shore protocol.

Finally, this report covers issues involved in the integration of voice/data in the NTS network for the years 2000 and beyond. Various voice/data integration schemes are examined with respect to overhead efficiency, error control, etc. No attempt has been made to determine the optimum voice/data integration strategy. The entire area of voice/data integration needs further analysis.

The key issues identified are asociated with switching and associated signaling techniques and presentation layer protocol, and include:

- Relative performance efficiency (in terms of in-band signaling versus out-of band signaling, this involves both the quantity of overhead bits and the handling of two types of traffic (information and control) with different characteristics (arrival distributions, priority, and length).

- The performance of these signaling schemes in a high error rate, dynamic environment; the use of forward error correction and separate header error detection to enhance performance.

- The complexities of co-ordinating the out-of-band signals (in the presence of errors) with the in-band transmission of information; this will involve both hardware and software co-ordination, possibly where the signaling and information networks have different topologies.

- The performance of reassembly algorithms for packet voice to provide intelligible conversation.

- The use of variable rate coding for voice digitization as flow control procedures; this involves dynamic algorithms where the coding is adapted depending on current traffic loads.

CONTEL
INFORMATION SYSTEMS

VOLUME I

COMMUNICATIONS REQUIREMENTS

AND

PROTOCOL STRUCTURE

1. INTRODUCTION

The goal of this task is to develop a protocol structure for future Navy Telecommunications System (NTS) which is being developed as a highly automated Command, Control, and Communications (C^3) system. A major requirement is that this system be sufficiently flexible to accommodate a wide variety of users with varied requirements for telecommunications such as real time tactical information exchange, bulk data transfer, and voice. It is clear that the NTS will be comprised of several communications systems or "networks," each of which will be optimized to meet a specific set of user requirements. These systems will operate over a wide range of the RF spectrum. It is clearly necessary to be able to switch traffic between these systems to enhance the reliability, survivability, and throughput of the total system. The protocol structure provides the mechanism for this interconnection.

The function of such a communication system is to support information transfer between communicating entities. A formal set of rules, understood and adhered to by the communicating entities is essential to the orderly transfer of information. Such a set of rules or procedures constitutes a protocol. In practice, the transfer of information via data communications requires the support of several functions to ensure reliability and efficiency. These functions can be logically grouped into "layers" or "levels." Each information source or destination is supported by an ordered layer of functions used for the exchange of data. The corresponding layer at a different site is referred to as a "peer" layer. Utilizing the services provided by its lower layer, each layer communicates with its peer layers according to rules or procedures which are defined as protocols. Interaction between a layer and its adjacent layers is subject to well-defined rules referred to as an "interface." To establish a convention, we refer to the layers in ascending sequence from the physical transmission media up to the end user.

The fundamental reasons for developing a layered protocol structure are interoperability, modularity, and tractability. Because of the many different systems deployed by the military, interoperability is clearly a major issue. To facilitate interconnection and interoperability, it is necessary that communications have a well defined, structured architecture, and that each subsystem and network conform to this common architecture. It must be emphasized that a common architecture does not imply that all subsystems and networks are identical; the architecture only enforces uniformity at the interfaces.

Second, by creating a layered structure, replacement and standardization of functions are facilitated. For example, if a function is to be replaced, only the layer containing that function must be modified (as long as the interfaces to the higher and lower layers are maintained). Also, this localization simplifies standardization because interfaces with external functions are minimized.

Clearly the computer communications networks, both currently being developed and envisioned for the future, employ sophisticated technology to meet broad sets of requirements. As a result, the communications protocols will be complex. Thus, to make a complex problem tractable (not only system interconnection but also system design), the functional specification of protocols is partitioned into layers. Because of the natural hierarchical processing requirement inherent in networks, layering is a natural partition. Since a layer will have interfaces with only its aojacent higher ana lower layers, the number of interfaces is minimized.

However, the layering required for system definition may be more granular than for system interconnection. For example, if the protocol architecture specifies additional layers, then all systems implementers must conform to the layers; this inherently constrains the implementer. However, in designing his individual protocol structure, the system implementer may want to introduce additional sublayers beyond what is standardized.

The technical approach pursued in the development of a protocol structure for NTS is to evaluate Open System Interconnection model developed by the International Standards Organization as a baseline structure. The ISO-OSI reference model is a layered architecture and consists of the following seven layers (Reference 1):

1. Physical Layer: provides the mechanical and electrical interfaces, timing, and synchronization for the transparent transfer of bits (information) across a communication channel.

2. Link Layer: provides for the transfer of data between adjacent nodes.

3. Network Layer: provides for the transfer of data between non-adjacent network nodes by using proper routing procedures.

4. Transport Layer: provides for the transparent transfer of data between session entities. "Transparent" refers to the fact that the details of the underlying network operations are "hidden" from the higher levels.

5. Session Layer: enables the presentation-entities to organize and synchronize their dialogue and maintain their data exchange.

6. Presentation Layer: provides the syntax for the information transfer to its application layer.

7. Application Layer: provides the semantics for information exchange from/to the application process.

In particular the focus of the study has been on the lower layers, 1 to 4. However, this reference model has been developed primarily for use in civilian public and private networks. Thus, it is necessary to assess whether special requirements of the NTS can be accommodated by the OSI model. These special requirements include:

- dynamic operating environment caused by node mobility (at varying platform speeds) ranging from subsonic to supersonic speeds), node destruction by hostile forces, and degraded link quality caused by jamming.

- diverse traffic characteristics with disparate traffic arrival rate distributions, message lengths, and performance requirements.

- multiple types of error control for both detection and correction.

- access and reception of a communications channel by multiple users (instead of point-to-point transmission/reception).

The definition of a protocol structure consists of the identification of the functional requirements, assignment of functions to layers, and assignment of layers to specific network components. The assignment of functions to layers includes the relative order of the layers.

The criteria for layering are thoroughly discussed by Zimmerman (Reference 2). In summary he recommends collecting similar functions in a layer, especially if standardization of such function is likely; creating layers consisting of functions that can be implemented via the same technology; collecting functions to minimize the interactions between layers; and allowing bypassing of layers (without necessarily introducing additional interface requirements); but in doing so, do not unnecessarily constrain the systems engineer or implementer. Of course, by introducing layers the systems engineer and implementer are inherently constrained. Thus, there is a fundamental tradeoff between introducing structure and constraining the system developer.

Another major issue associated with the assignment of layers to specific network components in the protocol structure is the distribution of intelligence. However, this must be addressed in terms of the specific network architectures considered. In this study the networks specifically considered are the:

- HF Intra Task Force Network
- Navy EHF Satellite Network (NESP)

In the following sections we first summarize generic communications requirements in Section 2 and then discuss the individual layers:

- Physical and Data Link Layers in Section 3.
- Network and Transport Layers in Section 4.
- Session, Presentation, and Application Layers in Section 5.

Since congestion control and security permeate all layers, these issues are discussed separately in Sections 6 and 7. In Section 8 we then discuss the impact of layering in system performance. Our conclusions are summarized in Section 9.

2. NTS TRAFFIC REQUIREMENTS

The future Naval Telecommunications System (NTS) will include several communications network systems operating over a broad frequency spectrum. The NTS will provide the communications resources to support the command and control of Navy platforms. The evolving NTS will exhibit some marked differences from the current systems. For example, the NTS will mark the transition from an analog to a digital communications mode wth enhanced security and jam resistance. There will be greater use of the now tested technique of packet switching, and perhaps some of the more novel approaches of integrated and hybrid networking. This section will provide a brief discussion of the traffic characteristics on the NTS. A very detailed characterization of Navy communications requirements in the post-1985 time frame is given in the "Trilab Study" (Reference 3). The Trilab Study provides quantitative data on several hundred Navy communications needlines. The information provided in the Trilab Study has to be understood and consolidated for use in the sizing and design of specific NTS networks. However, our goal in this memo is to categorize the needlines in this study in order to establish generic requirements for a protocol structure. This generic characterization of the traffic requirements is stated in terms of the arrival pattern, message length, error control, and volume.

The NTS traffic requirements can be classified into six major functional categories which have unique service requirements that must be accommodated by the NTS architecture. We discuss these service requirements below.

a. Record Traffic and Commands. This type of traffic is bursty and requires high reliability (i.e., sequenced, error-free delivery). These messages generally require acknowledgement at the transport level and at the user/application level. The transport level acknowledgement serves to inform the sender that the message has been delivered without error by the communications system. The user level acknowledgement informs the sender that the receiver entity is able to process and comply with the message. Traffic in this category will exhibit various levels of urgency from routine to flash. Most of the traffic will be fixed format.

b. Weapons Control and Guidance Traffic. Most of this traffic will be fixed format, real time data exchange for weapons control and missile guidance. This type of traffic is one way and requires error detection; error correction is not generally required since the control messages are refreshed very rapidly. For the same reason, acknowledgements are seldom required. This class of traffic is bursty and the peak data rate will be significantly higher than current net loadings. For example, the missile control net loading assuming a 160 missile volley from a task force is expected to be 75 Kbps (Reference 4). Another example of time critical control traffic relates to the Aircraft Carrier Landing System (ACLS). The carrier must transmit time critical control information to the approaching AC at an update rate of 10 messages (75 bits long) per second.

c. Surveillance. Surveillance traffic consists of hostile track reports and real time track management messages. This class of traffic may account for about 90 percent of the NTS traffic in a stressed environment. The track reports are generally fixed format messages and are broadcast to the entire community. These reports are refreshed periodically (typically once every 3 to 12 seconds) and, hence there is no requirement for retransmissions. Also in this category is the requirement to transmit raw sender data from platforms such as the S-3A for processing on board the host platform. The requirement for local sensor data transfer to host platforms is established to be 31 3 KHz channels (Reference 4) and for long-range ship-to-shore sensor data transfer, the estimate is 10^5 to 10^7 bps.

d. Position Location and Identification. This traffic consists of short periodic messages transmitted by all active platforms. The primary purpose of this traffic is for IFF; however, these messages generally have sufficient spare bits that can be used for conveying other platform related information.

e. Voice. The NTS must support digital voice communications. The requirements for voice circuits include:

- Two 16 Kbps circuits for each ocean area for shore-to-ship communications. This net should offer voice conferencing between shore based command entities and Task Group/Task Force commanders at the scene of action.

- 2.4 Kbps tactical voice circuits. Several of these circuits will be assigned to each platform, depending on its mission assignment. For example, a large amphibious assault platform would require about 16 2.4 Kbps voice circuits. These circuits are currently half-duplex, demand access channels relying on human protocols to control channel access. It is expected that the NTS will evolve more sophisticated access control and signalling schemes, and also explore the issue of integrated voice and data networks. Of course, the specific bit rate depends on the voice digitization rate to be employed.

f. Graphics/Digital Facsimile. This class of traffic is generated by such applications as high resolution photography from reconnaissance AC to CV. This application requires a 1.5 Mbps (5 Mbps desired) half-duplex channel to provide a 10 minute response time. Other non-real time applications for Digital Facsimile also exist and it is anticipated that these will be accommodated as low priority traffic within the data nets.

The high-level classification and characterization of the NTS traffic points out the need to develop a flexible network architecture that will support different protocols and provide a variety of services required for Navy command and control applications. The remaining sections of this memorandum explore the issue in greater detail.

3. PHYSICAL AND DATA LINK LAYERS

3.1 Functional Requirements

To operate in the military the functions of the physical and data link layers will be significantly different than corresponding layers in commercial networks. A hierarchical representation of sublayers for these functional requirements, depicted in Figure 1, consists of sublayers for

- link acknowledgment and sequencing
- addressing
- dumb relay
- access
- error detection
- error correction
- spread spectrum
- transmission and synchronization

In a typical commercial network, only the transmission and synchronization functions, addressing, link retransmission and sequencing, and error detection are required. However, in a military network spread spectrum, (forward) error correction, access, and dumb relay may be required. The dumb relay actually corresponds to a circuit switch in which the relay time and output port or net (if appropriate) are based on time and space (input port or net) of the received bit stream. An example of a dumb relay would be the paired slot relay in the Joint Tactical Information Distribution System (JTIDS). In this network the slot in which a message will be relayed is based upon the slot in which it is received. Also in the NESP satellite, the data communications links are circuit switched; hence the dumb relay sublayer will have to be included in that network protocol structure.

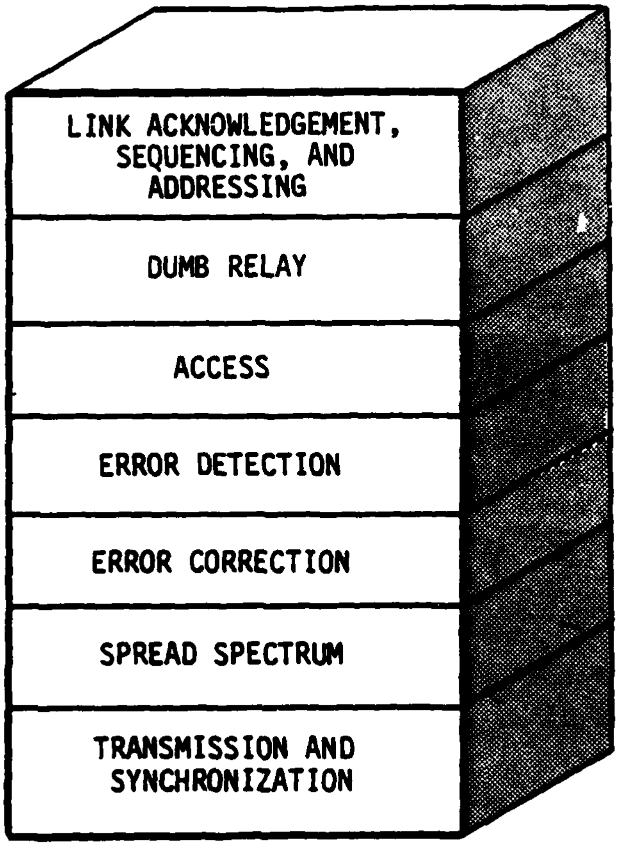

LINK ACKNOWLEDGEMENT, SEQUENCING, AND ADDRESSING
DUMB RELAY
ACCESS
ERROR DETECTION
ERROR CORRECTION
SPREAD SPECTRUM
TRANSMISSION AND SYNCHRONIZATION

FIGURE 1: SUBLAYERS FOR PHYSICAL AND DATA LINK

Because of the environment, there are many unique requirements. For example, even with the use of the sophisticated transmission techniques, spread spectrum, and forward error correction, it still may not be possible to guarantee communication. In fact, one way links may result in which unit A may hear unit B, but not vice versa. Thus, two-way communications may not be accommodated. As a result, the link transmission and sequencing in the data link layer will be different than commercial link protocols such as the high-level data link control (HDLC). In particular it may be preferable to delegate these accountability functions to higher level protocols, making the functional requirements of sequencing and retransmission vacuous.

Because of the dynamic environment, one of the major issues to be considered in the HF Intra-Task Force network using the cluster architecture is the requirement for reinitialization of the links as the clusters change. For example, if the cluster algorithm defines new backbone links, it may be required to initialize new links (set sequence numbers to zero, etc.) and possibly terminate some links. However, this may not be such a difficult problem if accountability is vested in the higher layers.

Also, in contrast to most commercial networks, it may be necessary to have multiple logical links between adjacent platforms. This may include separate links based on different user functional requirement (data or voice, etc.) and also may include a separate link for an orderwire; this is easily accommodated by the protocol structure. Furthermore, by employing separate links, priorities can be effected by the algorithm used in servicing them; i.e., the lower level protocols would service the links in a system specified order.

As discussed above, broadcast, group and function addressing must be incorporated into the protocol structure. Addressing is a natural part of the link retransmission and sequencing function and will require the capability to include group addressing in the header. Function address initially appears to be a significantly difficult problem to incorporate into the data link layer because function is associated with the application layer. However, at the link level this can be accommodated by employing group addressing corresponding to functional capability. Of course some platforms may have to be addressed by multiple group addresses.

Furthermore, functional addressing is a broader problem because the sending platform may not know a priori, the address of the receiving platforms having the requisite functional capability. In fact, the sending platform may not then know how many of the destination platforms exist. In this case, broadcasting will be required. To completely handle this problem, the functional capabilities will have to be incorporated into higher layers. For example, the network layer may be responsible for relaying a message (possibly on multiple links or nets) and forwarding it to the host to mininize the number of transmissions. Furthermore, the session layer may also forward the same message over multiple connections. All of the above options are compatible with the OSI architecture.

In summary, at a minimum the link level header will have to be able to handle a multi-destination transmission capability, i.e., transmission to multiple platforms simultaneously. Also, the network layer may have to be able to route the same packet to multiple destinations, i.e., more than one (but less than all) receiving platforms may have to relay the packet. The session layer will also have the requirement of identifying the requirements for multi-destination messages based on message types/user input, etc.

The error detection for a data link in the military will be somewhat more complex because it must be possible to apply the error control schemes selectively based on the traffic type. For example, voice messages and surveillance data which is periodically refreshed, should be subject to only error checking, but no correction by retransmission. However, this can be easily accommodated via separate links or possibly unnumbered frames.

The location of the error detection function depends on the technology used to implement it. For example, the Cyclical Redundancy Checksum (CRC) is the standard procedure used to detect errors. This may be implemented in either hardware or software; the current trend is to implement it in hardware. If it is implemented in hardware, it is natural to include detection in the physical layer; while if it is implemented in software, it is more appropriate to include it in the link layer. Thus, in a functional specification, it is appropriate to include it optimally in both levels; the inclusion in a specification can be made optional.

Error correction schemes must be adaptive. For example, as the link error rate increases due to jamming, control information should be passed to the link layers to adopt a more powerful error control procedure. Note this may require exchange of control information between the retransmission and sequencing sublayer, which identifies the need for increasing coding, and the forward error correction sublayer which effects the increase of coding. Furthermore, the receiving decoder may have to pass control messages to the transmitting coder indicating the code rate should be increased or decreased. This could be included for the header or a separate message could be passed via a transport mechanism. In the OSI architecture, the algorithms for adjusting the code rates can be viewed as monitor and control applications processes. These applications interface with the lower level protocols by using them for exchange of information and for passing control parameters to them.

One of the clever advantages of military networks operating in a broadcast environment is the use of the passive or echo acknowledgement; this is employed in the Packet Radio network (Reference 5). In this scheme, the originator of a message will listen for a relay transmission of a message and accept this as an acknowledgement. However, to detect such an acknowledgement requires the originator to examine network routing information (part of the network layer) to *identify a data link acknowledgement.* This violates the layered protocol structure which separates data link and network functions. For example, the link and network layers are no longer independent because a change in the network layer would impact the data link layer. To adhere to the protocol structure, redundant routing information could be placed in the data link layer. This additional overhead may outweigh the savings of the passive acknowledgement. This remains an issue for further performance evaluation.

Another major issue is the incorporation of the access function. The access layer is substantially different in function and may merit a separate layer. This is not an issue in commercial interfacility networks because point-to-point transmission facilities or multidrop lines are employed. However, the IEEE 802 Standard for Local Area Networks (Reference 6) has proposed separate sublayers for media access and data link control. The IEEE 802 includes both a Carrier Sense Multiple Access with Collision Detection (CSMA/CD) and a Token algorithm for access. Since Local Area Networks are also important in the military, a reasonable consideration for NTS is to define an access layer (including the dumb relay function) between the data link and physical layers. However, the implementation considerations of this must be considered.

3.2 Layering Analysis

Based upon these issues, it is recommended that only two layers be defined for standardization, namely a data link layer and an enhanced physical layer. The recommended assignment of functions to layers is depicted in Figure 2. The substance of this strategy is to merge all the functions intimately related to the communications channel into a single physical layer, but maintaining a data link layer. The data link layer is characterized by potentially requiring substantial processing and exchange of information (ACKs, NAKs, resets, etc.) with its peer layer; the physical layer only has minimal signaling (e.g., CSMA/CD) with its peer layer.

The rational for incorporating more functions into the physical layer is that

- these functions are likely to be implemented in the same technology.
- their characteristics are dependent on the communication channel.
- the assignment allows for the reordering of some functions within the physical layer.
- baselines for standardization of the sequencing and retransmission functions exist.

In particular, the hierarchical order of the layers presented in Figure 1 is implementation dependent. For example some implementations may want to use a microprocessor to provide the forward error correction to the frame, buffer it, and then transmit the frame. In this case the FEC layer would be above the access layer. Since there is no strong reason to prohibit this implementation, it is best to devise a general standard that will allow the system implementer to decide the relative hierarchy of the sublayers. Furthermore the requisite characteristics of the FEC algorithm are more dependent upon the properties of the communications channel than of the link and higher level functions. Thus it is appropriate that coding be included in the physical layer. Similarly the synchronization preamble transmitted before the basic frame may include some information about the FEC code rate. This argues for both access and FEC to be in the physical layer, and not to be broken out as separate layers.

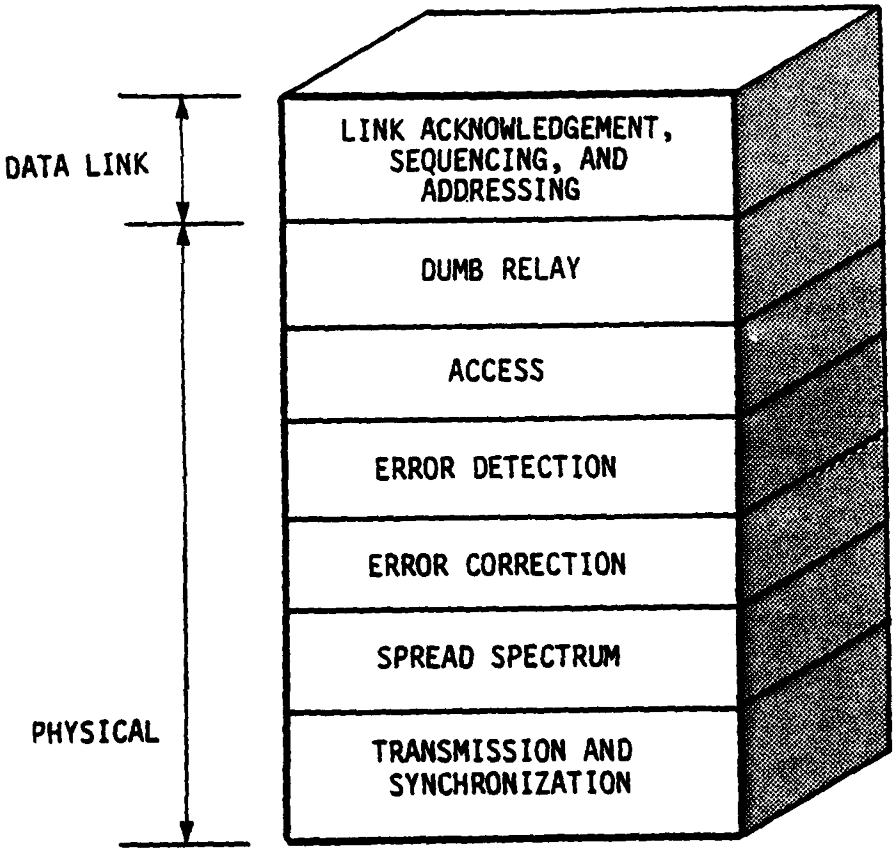

DATA LINK
PHYSICAL
LINK ACKNOWLEDGEMENT, SEQUENCING, AND ADDRESSING
DUMB RELAY
ACCESS
ERROR DETECTION
ERROR CORRECTION
SPREAD SPECTRUM
TRANSMISSION AND SYNCHRONIZATION

FIGURE 2: RECOMMENDED LAYERING

Note in this type of protocol architecture, the differences between the JTIDS TDMA and DTDMA structures are localized to the physical layer. However, these are substantial similarities within the physical layer which could be standardized within the JTIDS framework.

One of the disadvantages of this protocol architecture is that in a slotted access method, the impact of slotting will permeate higher layers, up to the message formats of the application level. Specifically to optimize network performance, we would want to optimize the message format to the slot size. However, this problem appears inevitable in all protocol structure. Regarding protocol residency, the link level protocol would reside in all ships in the task force as well as the NESP satellite. However, this is required in the NESP for reliable control of monitor and control information, not the exchange of information (voice and data), which is circuit switched, i.e., dumb relay.

4. NETWORK AND TRANSPORT LAYERS

4.1 Functional Requirements

Because of the hostile environment, e.g., potential component destruction, jamming, one way links, the physical and data link layers cannot be assumed to provide a reliable transport mechanism. Therefore, sophisticated protocols must be incorporated into the network and transport layers.

In particular a major requirement of the network layer is the capability of adapting to the dynamics of the military environment. Thus the network layer must be capable of dynamically monitoring the network state and altering routing parameters. The cluster algorithm for the HF ITF network is an example of such an algorithm. In addition to routing, the network layer would also accept segments from the transport layer and form packets as input to the link layer. The transport layer must be responsible for guaranteeing a reliable end-to-end transport mechanism; the Transmission Control Protool (TCP) (Reference 7) is a protocol meeting the requirements of the level. This possibly includes retransmission, sequencing, and reassembly of messages or packets. In particular the subtle issues of detecting duplicates, out of date messages, and out of sequence packets are very important issues in the military environment because of the unreliable lower level protocols.

In the proposed HF Intra-Task Force network, the network layer would recognize the clusters, assign routes according to the output of the cluster algorithm, and initiate links between platforms determined by the algorithm. The actual processing associated with the algorithm would run as a Monitor and Control application process, employing the lower level protocols as transport mechanisms. Cluster assignment, etc. is passed to the network layer as control information from the application layer.

The network and transport layers would be resident in the ships as well as the NESP. In particular the satellite would require the transport layer for the reliable exchange of control information; the network layer may be needed in the satellite for routing or packetization of such information or it may be vacuous.

Two of the fundamental issues associated with the network and transport layers in a packet switched network are virtual circuit versus datagram procedures and internetworking. However, both can be accommodated in the OSI architecture. In a virtual circuit routing procedure a path is set up before transmission of information. This reduces the amount of routing information that must be maintained in the header when information is exchanged. Although channel capacity will not be dedicated at setup, some resources may be assigned such as buffers. In datagrams routing individual packets/messages are routed through the network independently. In military networks the dynamics of the environment may introduce substantial overhead in maintaining (re-establishing) a virtual circuit path as units move or are destroyed and as jamming is introduced. Hence the reduction of header size and simplified routing associated with virtual circuits may be outweighed. Considering the dynamics of the cluster algorithm, datagram may be more appropriate than VC which would have to be reestablished when the clusters change.

The other major issue associated with these layers is where to place the internetwork layer. The internetwork function would include the routing between networks and requisite fragmentation and assembly. Given its importance in interconnection, the internetworking function deserves a separate layer. However, additional layers may be required. As depicted in Figure 3, to consider the internetworking problem in general, networking and transport functions must be considered for both intranetwork and internetwork. The Transmission Control Protocol and Internetwork Protocol (References 7 and 8), which have been adopted by the DoD as standards, are protocols that satisfy the internetwork transport internetwork layers.

4.2 Layering Analysis

It is recommended that the protocol architecture depicted in Figure 3 with separate layers for

- intranetwork
- intranetwork transport
- internetwork
- internetwork transport

be adopted.

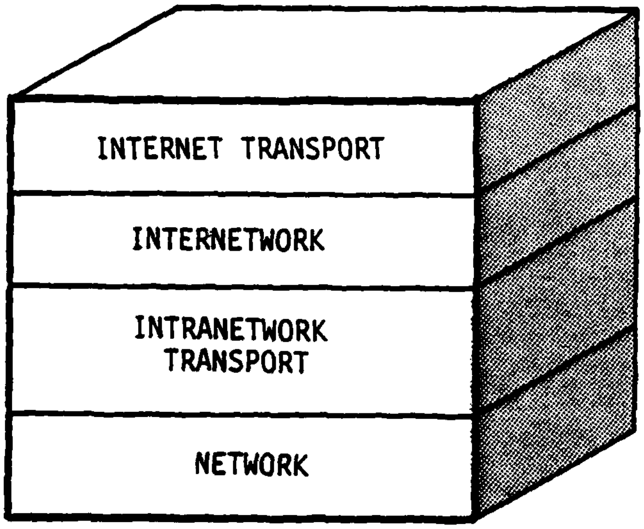


FIGURE 3: GENERALIZED NETWORK AND TRANSPORT LAYERS

The protocol architecture must be sufficiently general to include the layers depicted. In particular the partition between network and internetwork layers must be recognized. The intranetwork transport layer is included for completeness; it may in fact employ the same transport procedures (TCP) as the internetwork transport layer. However, this layer would be used only for intranetwork information exchange and would be vacuous for internetwork information exchange.

Another alternative is to employ the CCITT X.75 Standard (Reference 6) for the internetwork layer. This approach is to concatenate virtual circuits associated with each network, but it does not provide end-to-end accountability for the transport layer.

5. SESSION, PRESENTATION, AND APPLICATION LAYERS

5.1 Functional Requirements

In order for the user or application process to interface with the transport subsystem, several additional "higher level" functions are needed. We shall refer to these higher level functions collectively as "Session Services." In terms of the Open System Architecture, the Session Services encompass the session, presentation, and application layers.

The session layer is responsible for establishing, maintaining, and terminating session connections. Establishing a session connection may entail:

- Transport subsystem selection to identify the proper subset of NTS network and communications resources.
- Signalling to exchange control information such as channel ID, speed, priority, etc.
- Access control to ensure that a session binding is authorized and feasible.
- Transport Subsystem Service requirement. The session entity may request reliable end-to-end transport service from the Transmission Control Protocol or use a different transport protocol for supporting such applications as packet voice.
- Service/resource availability check.

In the military environment, a major requirement is session maintenance, which requires monitoring the Transport Subsystem (TS) performance and providing for transparent recovery from TS failures. For example, in the military environment, this may entail switching from a satellite network to a terrestrial network. In the NTS environment, this may entail switching from the improved HF network to JTIDS for transmitting tactical data. Of course, at the implementation level, we must specify what constitutes a TS failure, how it is detected, and what is the recovery procedure. Session maintenance may also require a pacing control function to regulate the timing of data transfer and is especially needed if one of the session entities is supporting data transfer to an electromechanical device. Of course, at the implementation level, we must specify what constitutes a TS failure, how it is detected, and what is the recovery procedure.

As noted above, it is necessary to incorporate priority mechanism in the protocol structure. It is preferable to implement this at a higher level such as the session level because the priority selection criteria is primarily dependent upon the application rather than the attributes of the lower level protocols. However, this requires a more detailed performance study because the higher in the protocol structure that prioritization is effected, the larger the latency delays become, i.e., once the prioritization is done and information passed to a lower level, the prioritization cannot be changed if the lower level does not have a prioritization capability.

The presentation layer provides the semantics for the information transfer. It provides the function necessary for format conversion, data transformation including encryption and decryption, code set translation, and data compaction and expansion. Since a diversity of terminal and host types are expected in the military environment, the presentation function merits a separate layer.

The applications layer provides the interface to the user processes. The functional capabilities of this layer are very much dependent on the nature of the user processes supported. We give some example of applications layer protocol that will be required in the NTS:

- Remote Batch Processing Protocols: to handle raw sensor data transfer and processing.

- Data Base Exchange and Synchronization Protocols: to support such applications as relief or replacement of an active E-2C.

- Process Control Protocols: to transfer real time control information for weapons control and missile guidance.

- Image and Graphics Protocols: to transfer sensor data to pictorially display friendly forces and hostile threat.

- Net Management Protocols: to support network monitoring and recovery functions.

Some of these protocols will be sufficiently general to require system-wide standardization. Others will be standardized within specific subsystems such as a weapons subsystem component interfacing to the NTS.

5.2 Layering Analysis

The OSI layering is sufficiently general to accommodate the functional requirements discussed. Hence it is recommended this be adopted.

6. CONGESTION CONTROL

Congestion control, which permeates several protocol layers, refers to the procedures used to regulate the message traffic in a network so as to prevent or alleviate network congestion. Most congestion control algorithms perform other related functions such as allocating network resources fairly, prevention of message "deadlocks," and controlling the speed with which a host can transmit/receive data. Congestion control may be applied at the link, network, or transport level. When applied at the transport level, it is known as "flow control" and serves to prevent congestion of user buffers at the process level.

In this section, we shall briefly discuss the control mechanisms in the various layers of the protocol hierarchy.

6.1 Link Congestion Control

This level of control regulates the traffic between two neighboring, connected nodes in order to prevent local buffer congestion and deadlocks. Although the control is applied locally, there could be end-to-end repercussions due to the "back pressure" effect (i.e., the propagation of the blocking condition upstream to the traffic source).

The two main issues relating to link level congestion control are:

1. Parameter to be monitored for detecting congestion.

2. Procedure to control the flow of data.

The most common parameter used to monitor congestion is buffer occupancy. Buffers may be partitioned into classes based on different criteria such as output channel, virtual call number, traffic class, etc. The buffer occupancy of each class is then monitored to regulate the traffic and enforce a fair sharing of the node resources.

The common techniques used to control the flow are:

1. Dropping packets (which are then retransmitted by the originator).

2. Window scheme in which the transmitter maintains a window of sequence numbers that are allowed to be transmitted on each link. The receiving terminal controls the window by sending control messages which are generally included in the link ack. The receiver can also send a control message indicating it is not ready to receive any data.

6.2 Network Congestion Control

We distinguish between two types of network congestion controls:

1. Network Access Control: which regulates the ıraffic entering the network.

2. Entry-to-Exit (ETE) Control: which prevents buffer congestion and deadlocks at the destination network node.

ETE congestion control is most commonly implemented by means of a window scheme. The window size may be fixed or dynamically controlled; analytical queueing network models or simulation is generally used to determine the optimum window size.

Several different network access control schemes have been studied and implemented. The three major schemes that have been implemented are:

1. The Isarithmic Scheme, proposed by the National Physical Laboratories in English. This is a permit scheme which constraints the maximum network traffic to a constant. A fixed number of permits are circulated in the network, and a packet must acquire a permit before being allowed to enter the network for transport. The Isarithmic scheme is a global control mechanism (Reference 9).

2. The Input Buffer Limit scheme is a local throttling mechanism based on the buffer occupancy at the network entry node. This scheme has been implemented in the German GMD network (Reference 10).

3. The Choke Packet scheme, proposed for the French Cyclades Network, is based on sending small control ("choke") packets to the source node whenever traffic from that source encounters a congested route (Reference 11).

6.3 Tranport Layer Congestion Control

The function of the transport layer flow control is to prevent congestion of destination buffers. It also serves to prevent message reassembly deadlocks by reserving the required buffer space. Transport layer flow control is generally implemented using a window scheme in which the receiver sends credits to the transmitter authorizing transmission. The credits are sent as control information and each credit specifies a message sequence number, N, and a window size, W. Upon receiving the credit, the transmitter is authorized to transmit all messages up to the message bearing the sequence number, N + W. The received credit also serves as an ETE ack for all messages up to sequence number, N.

6.4 Congestion Control in the NTS

Having surveyed the well known congestion control strategies, we now focus on specific congestion control issues relevant to the NTS. The significant issues pertaining to the NTS architecture are:

1. If the NTS adopts the DoD standard Transmission Control Protocol (TCP), then ETE congestion control can be incorporated via the TCP retransmission algorithm. However, good retransmission schemes must be developed to ensure that the network does not become unstable due to excessive retransmissions (typical of the Aloha type broadcast radio nets).

2. The Network Access control schemes should be sensitive to the user's specification of traffic priority, i.e., as congestion begins to build lower priority traffic should be throttled and high priority traffic should be transported through the network.

3. Basic research and development work needs to be done to determine the proper congestion control strategies to be used in hybrid networks that use circuit and packet switching and in integrated networks that packet-switch voice and data traffic. For example, the congestion control scheme must take into account the fact that voice cannot be buffered and delayed in case of congestion and it is therefore better to block a call at the source when congestion is imminent.

4. It was pointed out at the beginning of this section that congestion control permeates several layers of the network architecture. Any implementation must carefully study the consistency and performance impact of the entire hierarchy of controls.

5. Another important congestion control mechanism relevant to the NTS environment and which requires further analysis is that of information "filtering," whereby the network entity detects congestion and sends control information to a higher level process to compact or edit the information contents of each messages.

6. The ETE congestion control that is implemented by a window mechanism must be modified to account for jamming that will prevent a terminal from receiving a permit to advance its window. The modified scheme should allow a jammed terminal to disable the congestion control window mechanism and continue its transmission.

7. SECURITY REQUIREMENTS

Security in NTS is a vast subject and a detailed study is beyond the scope of this task. However, the protocol structure provides a basic mechanism to invoke security measures. Because of the hostile environment in which military networks are deployed, the security requirements are a major factor in the development of a communication network, and their functional requirements permeate all protocol layers. The primary functional requirements consist of encryption of data, encryption of communications control information for each protocol layer, and random generation of pseudo noise (PN) codes, frequency hopping patterns, and associated synchronization control (jitter, sync patterns); this is commonly referred to as transmission security. However, the *transmission security is contained* in the physical layer while the encryption security requirements span all protocol levels. Furthermore, different key variables may be employed for function or layer.

The major issues associated with developing a security architecture are assignments of key variables to users and functions; number of users holding each variable; number of crypto variables to be managed by the security controlling authority; hardware/software complexity of interfacing a terminal to a crypto device; and protection provided by the security mechanism in terms of preventing a compromise and limiting the impact of a compromise.

Another function that must be accommodated by a protocol structure for the military environment is electronic key distribution; it is accommodated in this protocol structure in the application layer. This is similar to a monitor and control application. To distribute keys and associated status/advisory/command information (both in response to a compromise or for a normal update), the key distribution application will invoke the lower level protocols for reliable exchange of information.

At a minimum the security protocol architecture will have to include:

1. Internetwork end-to-end encryption for the user data and higher level protocols.

2. Intranetwork end-to-end encryption for user data and higher level protocol.

3. Link encryption for lower level protocols and transmission security.

These requirements can easily be accommodated in the layered protocol structure discussed above. The protocol structure provides the framework transec, data and control encryption for individual layers, and distribution of keys; the mechanisms for doing so are implementation issues. In particular key management is a critical issue because of the comple๑ity of distributing keys to a large number of users. In particular, users may have to communicate with a large number of users and it may not be able to store keys for each individual user. This may require having an intermediate gateway perform crypto variable translation or allowing many users without a need to know receive and decrypt the message.

8. PERFORMANCE

In the sections above, it has been discussed how the networking requirements for a military communications network can be incorporated into the ISO Layered Architecture. However, we must be concerned about the inherent efficiencies (or inefficiencies) that may be introduced by having so many layers. For example, perhaps the ISO structure is too general. To fully answer this question would entail the detailed specification of each protocol layer and an evaluation of their impact on performance, which is beyond the scope of this study. However, some general observations can be made.

Zimmerman (Reference 2) has enumerated some basic principles for defining the ISO layers as follows:

"1. Do not create so many layers as to make difficult the system engineering task describing and integrating these layers.

2. Create a boundary at a point where the services description can be small and the number of interactions across the boundary is minimized.

3. Create separate layers to handle functions which are manifestly different in the process performed or the technology involved.

4. Collect similar functions into the same layer.

5. Select boundaries at a point which past experience has demonstrated to be successful.

6. Create a layer of easily localized functions so that the layer could be totally redesigned and its protocols changed in a major way to take advantages of new advances in architectural, hardware, or software technology without changing the services and interfaces with the adjacent layers.

7. Create a boundary where it may be useful at some point in time to have the corresponding interface standardized.

8. Create a layer when there is a need for a different level of abstraction in the handling of data, e.g., morphology, syntax, semantics.

9. Enable changes of functions or protocols within a layer without affecting the other layers.

10. Create for each layer interfaces with its upper and lower layer only.

11. Create further subgrouping and organization of functions to form sublayers within a layer in cases where distinct communication services need it.

12. Create, where needed, two or more sublayers with a common, and therefore minimum, functionality to allow interface operation with adjacent layers.

13. Allow bypassing of sublayers."

These principles apply in the specific Navy military environment as Zimmerman describes for the general environment. Principles 3, 5, and 8 point to the justification of separate physical and data link layers. Since the expected Navy deployments indicate the requirement for multi-hop routing, network oriented protocols for routing and end-to-end accountability will be required. Hence, principles 3, 5, 7, and 8 argue for the introduction of the network and transport layers.

Management of the transport functions for different applications indicates requirement for a session layer while the possibility of a wide range of terminal types indicates the requirement for a presentation layer. Principles 3 and 4 justify creation of separate layers. Furthermore, principles 4 and 5 argue for separating communications and applications; hence, a separate applications layer is introduced.

When the functional capabilities of the network protocols are implemented, there will be a mapping of functional capability to computer program components. The components will exchange data via well defined interfaces. It is acceptable that a single layer may be implemented via multiple software components, e.g., Link Receive and Link Transmit. However, mapping of multiple layers into a single component reduces program modularity and defeats a fundamental principle of layering: localization of data. To facilitate modification of the functional capability of a protocol layer, it is desirable to localize the functional capability of a layer to a specific component. Specifically, if we are modifying the internals of a protocol layer, all other layers should remain unaffected. If the interfaces of a layer are being modified, then only the interfaces of adjacent layers should be affected.

There are alternative procedures for layers to exchange data:

1. Direct subroutine calls.

2. Mail boxes, where module will be periodically invoked to check if it has data to process.

3. Wakeup/Demand Scheduling, where a module in one layer will request that another module be restarted or scheduled to process data from the former module.

See Reference 12 for a more detailed discussion.

Thus, the overall impact of the number of layers will be very much dependent on how the layers exchange data. If it is done by a subroutine call, very little overhead will be introduced. However, this may not always be possible or practical. In particular, it is more likely information may have to be exchanged by invoking the mailbox, wakeup, or demand procedures, which will require the services of the operating system. Clearly, this will introduce substantially more overhead; this has been studied in Reference 13. Of course, in the military environment, the operating system can be tailored to optimize performance for military requirements.

For example, the amount of overhead can be relatively large when the amount of processing is small. Consider routing along the path of a virtual circuit. After a link receive function has sent a packet to the network layer, only a table lookup is required to route the message. Hence, it is tempting to bury the table lookup in the link receive component or (worse yet) a link receive subprogram.

Another important issue is the distribution of intelligence associated with allocating protocol layers to processor. For example, lower level protocols (level 3 and below) may reside in a communication processor while the higher level protocols (level 4 and above) may reside in a host. For example, it is difficult to implement TCP (and the associated lower level protocols) in the current generation of 8 bit microprocessors. This introduces a major complexity because typically the communications system and host applications system are not replaced or modernized simultaneously. For example, lower level protocols could be in the communications processor and the higher level protocols in the host. Synchronizing these implementation schedules may not be easy.

9. CONCLUSION

In summary, our assessment is that a layered protocol structure is necessary for military communications networks and the ISO Reference Model can essentially accommodate military requirements if suitable generalizations are employed. Specifically additional sublayers should be broken out for both system interconnection and design. The only unresolved problem is implementing echo acknowledgements.

However, there are several caveats to be stated regarding a layered protocol structure. First, sufficient flexibility must be built into the protocol structure to allow bypassing of layers, primarily for distribution of control information. Bypassing of layers may introduce additional interfaces, which should be minimized.

Second, although the functional design of different layers can be done independently, there can be severe impact on performance if the characteristics of different layers are not considered. For example, in a slotted transmission architecture, the design of message formats must consider constraints imposed by the slot length.

Another major issue associated with a layered protocol structure is the impact on the performance of the layering. The impact on the network processor consists of the overhead associated with the exchange of information between layers. The extent of this impact is largely dependent on how the functions associated with the layers and their interfaces are implemented in the processor. Typically this will involve the invocation of the operating system and the associated overhead. Hence it is undesirable to invoke the operating system unnecessarily, but rather a clear partition between layers must be effected, i.e., the interface should not be buried in the middle of a complex module.

REFERENCES

1. Reference Model for Open System Interconnection, ISO, June 1979.

2. H. Zimmerman, "OSI Reference Model - The ISO Model of Architecture for Open Systems Interconnection," IEEE Transactions on Communications, April 1980.

3. Post 1985 Naval C^3 Requirement Study — Trilab Effort.

4. Naval C^3 System Design Principles and Concepts, August 1976. Naval Electronics Laboratory Center, San Diego.

5. R. Kahn et. al., "Advances in Packet Radio Technology," Proc. IEEE, Vol. 66, pp. 1468-1496, November 1978.

6. A. Tanenbaum, Computer Networks, Prentice Hall, Englewood Cliffs, NJ, 1981.

7. DoD Standard Transmission Control Protocol, Defense Advanced Research Projects Agency (DARPA), January 1980.

8. DoD Standard, Internet Protocol, DARPA, January 1980.

9. D.W. Davies, "The Control of Congestion in Packet-Switching Networks," IEEE Transactions on Communication, June 1972.

10. A. Giessler, et. al., "Free Buffer Allocation — An Investigation by Simulation," Computer Networks, Volume 2, 1978.

11. J. C. Majithia, et. al., "Experiments in Congestion Control Techniques," Proc. Int. Symp. Flow Control Computer Networks, February 1979.

12. J. Stankovic, "Software Communications Mechanisms: Procedure Calls Versus Messages," Computer, April 1982.

13. S.R. Bunch and J.D. Day, "Control Structure Overhead in TCP," Trends and Applications Proceedings, 1980, NBS.

CONTEL
INFORMATION SYSTEMS

VOLUME II

HF SHIP-SHORE PROTOCOL DESIGN

1. OVERVIEW

1.1 Introduction

In this volume the design of the High Frequency (HF) ship-shore communications protocol is presented. The methodology employed in the development of this protocol was to first review the performance requirements expected of the protocol and the environment in which the protocol would operate. Then the protocol design problem was structured into four building blocks: access/capacity assignment, link control, adaptive code rate, and net entry. The results of this methodology used for structuring the problem are summarized in Sections 1.2, 1.3, and 1.4. In Section 1.5, the performance metrics employed for evaluating alternative protocols are defined.

Then in the remaining sections of this volume the detailed specification of the protocol is described. In Section 2 a description of the system is presented, and then in Section 3 the frames employed in the protocol are defined. Then the protocol building blocks are described. First, the net entry protocol is specified in Section 4 and a summary of the Data Exchange protocol is presented in Section 5. In Sections 6, 7, and 8, the access, link, and adaptive code rate protocols are discussed. In conclusion, the issues of flow control are discussed in Section 9. Appendix A contains a list of the protocol specific parameters identified in this specification.

1.2. Requirements

Traffic Requirements

Although the characteristics of the traffic offered to the network by the users were not quantified, they were qualitatively addressed in terms of message arrival rate, message arrival distribution, and message length. It was assumed that the traffic arrival pattern was random, the message length was relatively long (hundred to thousand characters) and that messages had to be serviced by priority.

Intermittent Ship Arrival/Departure

Ships are mobile and will be intermittently entering and leaving the network. The geographical coverage of the network may span thousands of miles from shore.

Error Control

The network is responsible for the correct message between stations. Thus if a message is received incorrectly, the network is responsible for automatic retransmission. Cyclic redundancy checksums (CRC) will be employed to detect errors, and a very low undetected error-rate is expected. However, the design of the CRC is not an issue in this study.

Transparent Operation

The operation of the network should be automatic and transparent to the user to the maximum extent possible. It is desirable that the network be bit transparent in that any sequence of bits can be transmitted through the network. In particular, there should be no control characters that cannot appear in the bit stream, i.e., a bit oriented protocol rather than a character oriented protocol is required.

Radio Silent Mode

In emergency situations ships may enter a Radio Silent Mode in which the ship will not transmit messages (including acknowledgements), but may wish to receive messages. Hence, rather than depart the net, it will enter a receive only mode.

1.3. Environment

Transmission Channel

The HF spectrum will be employed as the transmission channel. In the full duplex mode of operation separate frequencies will be employed ship to shore and shore to ship. In the half duplex mode of operation, a single frequency will be employed for both ship to shore and shore to ship. The protocol assumes fixed frequencies, but will provide performance statistics to facilitate the operator in deciding when to manually change frequencies.

USQ 83 Digital Modem

The USQ 83 Digital Modem will be employed at 2400 bps for performing the modulation/demodulation functions. Sync time is estimated 0.8 seconds, but after sync is acquired it can be maintained indefinitely providing transmission is continuous.

Although the USQ 83 Modem is not intended to operate in the antijamming environment, it can operate at variable code rates. The code rate (CR) is the ratio of number of information bits transferred to the total number of bits transferred on the radio channel. Thus the information rate (IR):

$$IR = CR \cdot 2400$$

The following list defines the set of code rates and associated information rates:

Code Rate	Information Rate
1	2400 bps
1/2	1200 bps
1/4	600 bps
1/8	300 bps

At the current time the code rate on the USQ 83 modem is only manually adjustable, but it is assumed that the capability of adjusting the code rate automatically under computer control will be made available. To facilitate the adjustment of the code rate, the USQ 83 will provide a metric indicating the success of the decoding process. Whenever a transmission occurs, there is sufficient information in the preamble to identify the code rate. Hence, no additional information must be transmitted when the code rate is changed.

KG 84 Crypto

The KG 84 crypto, operating in the message indicator (MI) mode, will be employed for encrypting all message content (information text as well as headers). Thus all messages from shore (whether addressed to ship or not) must be decrypted and passed to the red side for address recognition. Sync time for the crypto preamble is 0.8 seconds. The same cryptovariable will be used by all users on a net.

Geographic Distribution

The shore station will communicate with ships over distances spanning thousands of miles. Hence it cannot be assumed that any pair of ships can communicate directly; this reduces the usefulness of multiple access techniques employing carrier sense.

1.4 Performance Criteria

In evaluating alternative protocols, the following criteria were employed:

Delays

The following delay criteria are defined for the network:

- Latency: time from the instant a message is ready to be transmitted until the instant transmission begins.

- Message Delay: time from the instant a message is ready to be transmitted until instant that the last bit of the message is received at the destination (by priority, 2 minutes for high priority).

- Access Time: time from the instant a ship desires to enter a net until the instant it is allowed to transmit (4 minutes).

Equitable Capacity Allocation

The protocol for ship to shore communication should allocate capacity in a manner that a ship with large requirements completely precludes other ships from delivering their messages.

Robustness

The protocol must be robust in the presence of transmission errors and other error conditions such that system crashes are not caused by such conditions.

Graceful Degradation

When peak loads are experienced, the network performance should not sharply deteriorate.

1.5 Problem Structure

As depicted in Figure 1, the protocol design problem can be partitioned into access/capacity allocation, link control, adaptive code rate, and net entry building blocks.

In this design, the access and data link level protocol functions of the ship-shore protocol will be developed. As necessary, assumptions regarding the adjacent (higher and lower) levels of protocol will be made.

Multiple Access

Since the traffic characteristics from ship to shore appear to be relatively steady and have a relatively loose delay requirement, the following multiple access techniques will be considered for data exchange:

- Time Division Multiple Access (TDMA) in which a fixed time period is assigned to each user (at net entry).

- Demand Assignment Multiple Access (Reservation) in which the shore station will dynamically assign capacity based on the current traffic requirements of each user.

- Polling in which the shore station individually sends a message to each ship when the ship is invited to transmit.

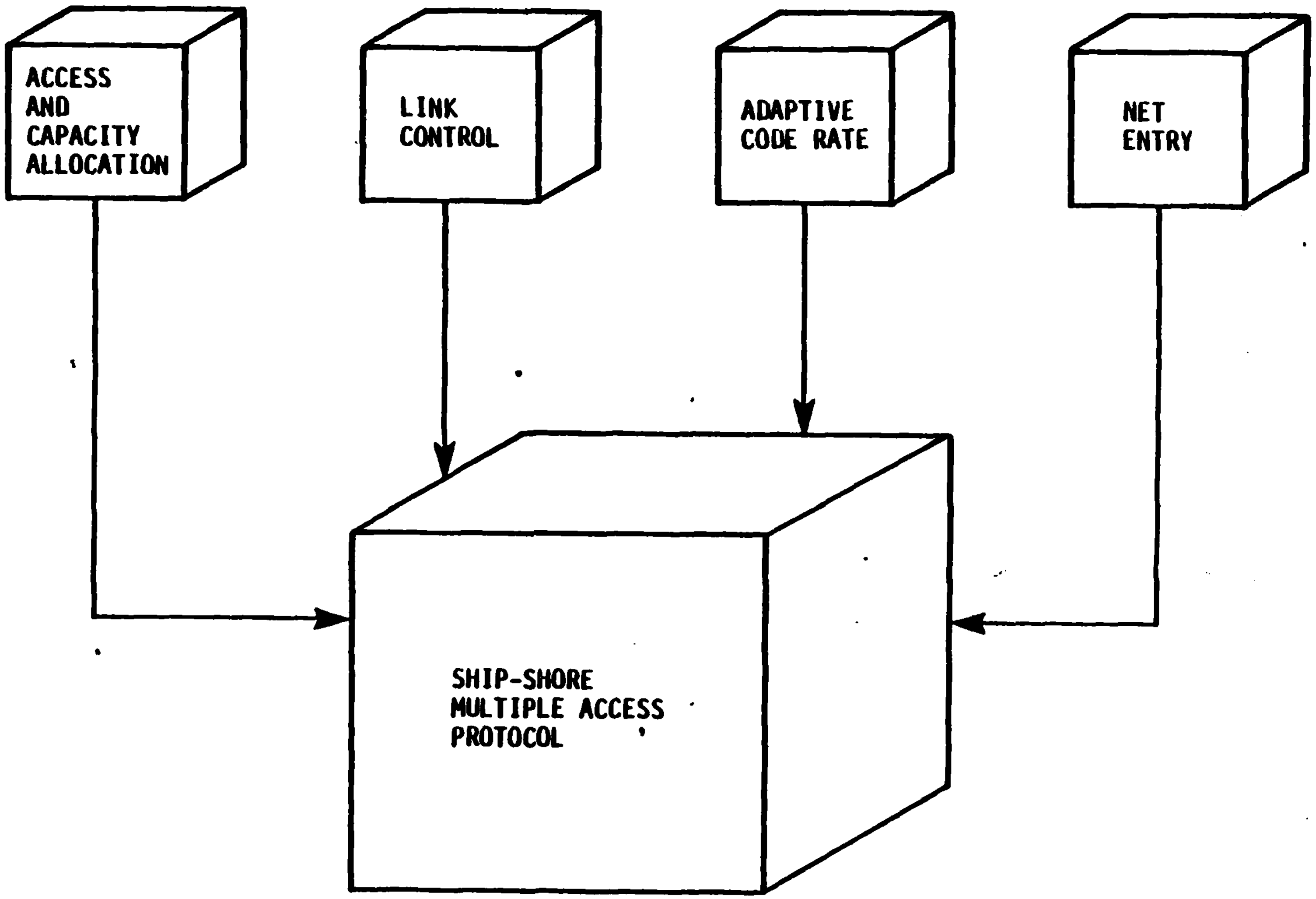


FIGURE 1. PROTOCOL BUILDING BLOCKS

An issue closely associated with the multiple access technique is capacity allocation. The important aspects of these issues are:

- Quantum for allocation such as time, blocks, or message.
- Time period for reallocating.
- Algorithm for deciding how much capacity each ship is provided.

These alternatives were carefully evaluated and a hybrid demand assignment and polling technique was designed. This scheme has the robust performance of a polling scheme in the high error rate environment and also the equitable distribution of capacity of a demand assignment scheme.

Adaptive Code Rate Algorithm

As described above, it is assumed that the USQ 83 can automatically adjust its code rate under computer control without incurring any additional overhead. The basic issues to consider:

- Algorithm for deciding when to increase code rate; this will be a function of the USQ 83 decoding metric and the rate of retransmission.
- Threshold for increasing the code rate.
- Whether decision is made locally (ship or shore) or centally (share).
- The same issues for decreasing code rate.

Note the ship to shore code rate may be different from the shore to ship code rate; in addition the code rate from shore to ship may be different for different ships.

After evaluation of this requirement, it was decided that manual selection of code rates was preferable because the environmental factors affecting performance change slowly. However, this is an issue for further study.

Link Control Procedure

Since the network is responsible for guaranteeing a very low undetected bit error rate, an Automatic Repeat Request (ARQ) Link Control Procedure will have to be incorporated into the protocol. The fundamental alternatives for ARQ are in increasing order of transmission efficiency:

- Stop and wait.
- Go back N.
- Selective repeat.

Most new protocols employ the go back N strategy since its sequencing and acknowledgement procedures are significantly simpler than the selective repeat algorithm.

Major aspects of this issue that must be considered are:

- Block size.
- Transmission time for acknowledgements.

Specifically, acknowledgements can be treated as individual and multiplexed with data messages, piggybacked onto data messages, and/or provided its own dedicated transmission capacity. Piggybacking of acknowledgements appears to provide significant savings with a modicum of overhead.

Furthermore, it must be possible to turn off the ARQ and adaptive FEC algorithm when a ship goes radio silent. In this case the shore would continue transmitting to the ship in the lowest code rate.

Because of the high error rate in the HF environment, the selective repeat protocol was chosen.

Net Entry/Exit

Since ships are mobile, they will necessarily be intermittently entering and departing nets. Thus a protocol is required to:

- Initialize the sequencing for the link protocol.

- Update a polling list and be incorporated into the capacity assignment algorithm.

In order to effect such an algorithm, capacity must be allocated for the transmission of net entry messages. This can be done in terms of a general poll or random access slots can be allocated. In any case because the shore may not know all ships that may want to enter, it appears inevitable that message collisions can occur at net entry. Hence a contention resolution algorithm must be incorporated into the algorithm such as random retransmission or tree searching.

Similarly, a procedure for orderly exiting a net (i.e., disconnecting the link) is required when a ship is departing the area of coverage of the shore station and/or has completed sending its message. A major issue to consider is whether in emergency conditions there is sufficient time to effect an orderly disconnection.

For this requirement, a slotted Aloha algorithm was devised in which the number slots used for net entry is dynamically adjusted.

In some cases, a ship may desire to enter another network even though it is already in one. One motivation for this is congestion control and load balancing. The basic issues are to:

- Identify net(s) that are overloaded.

- Determine ships that can be offloaded to another net without (significantly) degrading performance.

- Effecting the switchover from one net to the other.

However, this issue was not addressed in the study.

2. SYSTEM DESCRIPTION

The general problem is to enable multiple ships in a large geographical area (e.g., thousands of kilometers) to communicate with an onshore Network Control Station (NCS). The ship-shore communication system is illustrated in Figure 2.

The communications medium is a high frequency radio channel, which is characterized by high error rates. The modems at the NCS and the ships contain a CODER/DECODER. These serve to compensate for the high error rate characteristic of the medium. The CODER/DECODER performs this function by varying the code rate. Thus when the error rate of the medium is high, the CODER/DECODER should reduce the code rate and vice versa. A low code rate implies a low information transfer rate across the medium. The information rate will range from 300 bps to 2400 bps (the transmission rate is always 2400 bps).

Transmissions between the ships and the NCS occurs at several different frequencies. In the full duplex mode half of these frequencies are allocated for transmissions from the NCS to the ships, while the remaining half are allocated for transmissions from the ships to the NCS. The NCS and ships may selectively choose the frequencies on which they listen according to manual input.

This frequency allocation allows for full duplex operation of the data link. The half-duplex mode is not specifically addressed in this memo. However, its operation would be similar and less complex. In particular, the complexity is less because it is not necessary to address the simultaneous ship to shore and shore to ship transmission. Instead the NCS is essentially just another ship although having a larger, higher priority transmission load.

As seen from Figure 2, each station (at the NCS or the ships) contains a modem and an encryption device. There is a long synchronization time for the synchronization of the modems and the encryption devices at the sender and receiver. The synchronization information is carried in the preamble. This preamble precedes all transmissions to the receiver.

The NCS serves as the primary station while the ships behave like secondary stations. A major function of the NCS is to control all communications between the NCS and the ships. Communications between the NCS and the ships occurs in two phases. Ships desiring to enter the network are allowed to do so during the Network Entry Phase. A ship, initialized in the network, is allowed to communicate with the NCS only during its Data Exchange Phase. These two phases share the transmission capacity of the radio channel.

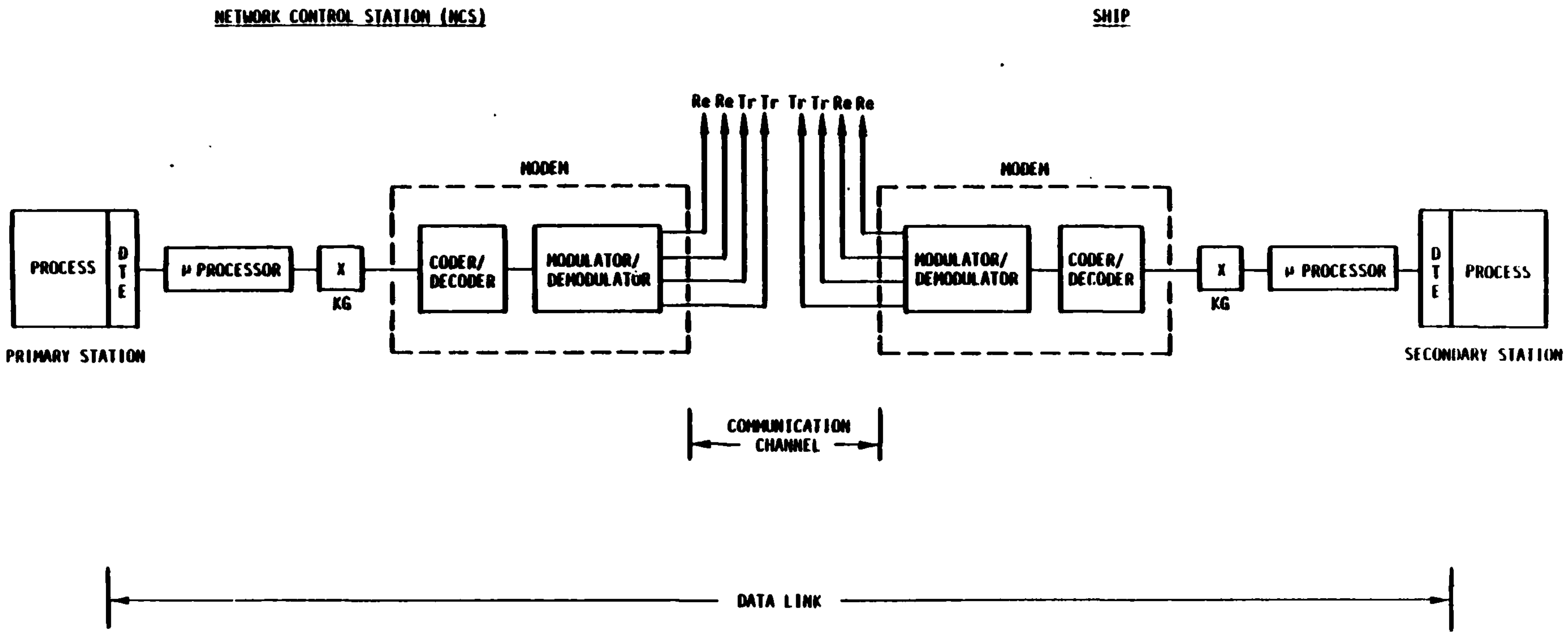


FIGURE 2. SHIP-SHORE COMMUNICATION SYSTEM

The Network Entry Phase generates a request for its initiation at fixed intervals of time. This request is honored only after completion of transmission of any frame to a ship. At this point the information rate is reduced to 300 bps. The Network Entry Phase messages are interleaved with the Data Exchange Phase messages with the Network Entry Phase messages having a higher priority. At the end of the Network Entry Phase, the information rate goes back to the information rate of the currently polled ship.

The polling scheme is used to grant access to ships, requiring to communicate with the NCS. All the ships in the network are grouped according to the code rate at which they are able to communicate with the NCS. The ships in a group are not necessarily in physical proximity to each other. Thus, all ships which can communicate with the NCS at an information rate of 300 bps are logically grouped together. The low code rate ships are polled prior to the higher code rate ships. Ships within a group are polled round robin.

Once a ship is granted access to the NCS, it has entered the Data Exchange Phase and full duplex communications occurs between the two. The selective repeat scheme is used to perform link control during the full duplex communication between the NCS and a ship.

A poll cycle is defined as the period of time required to poll, once, all the ships in the network. The relationship between the poll cycle, the Data Exchange Phase and the Network Entry Phase is illustrated in Figure 3. A poll cycle may contain only the Data Exchange Phase or it may contain both the Data Exchange Phase and the Network Entry Phase. The duration of the Data Exchange Phase is composed of the duration of the Data Exchange Phase at each of the polled ships. This duration is not a constant, but will depend on the number of ships polled and the quantity of data exchanged between the NCS and the ships. Because of the full duplex nature of the communication channel, the NCS can send data to ships already initialized in the net when ships desiring to enter the net are transmitting net entry requests.

The Network Entry Phase is initiated at fixed time intervals (T_{NEP}) asynchronous to the poll cycles. Thus it may begin at any time within the Poll Cycle. The Network Entry Phase need not be of constant duration.

Ships desiring to exit the network initiate such requests during their Data Exchange Phases. The NCS, on receipt of these requests, removes the ships from the network and the ships will no longer be polled.

Messages, sent between the application levels at the NCS and the ships, have a priority associated with them. The higher priority messages are transmitted before those of a lower priority. Message priorities are handled at the network level, a level higher than described in this report. Message priorities come into play only after a link is established between the

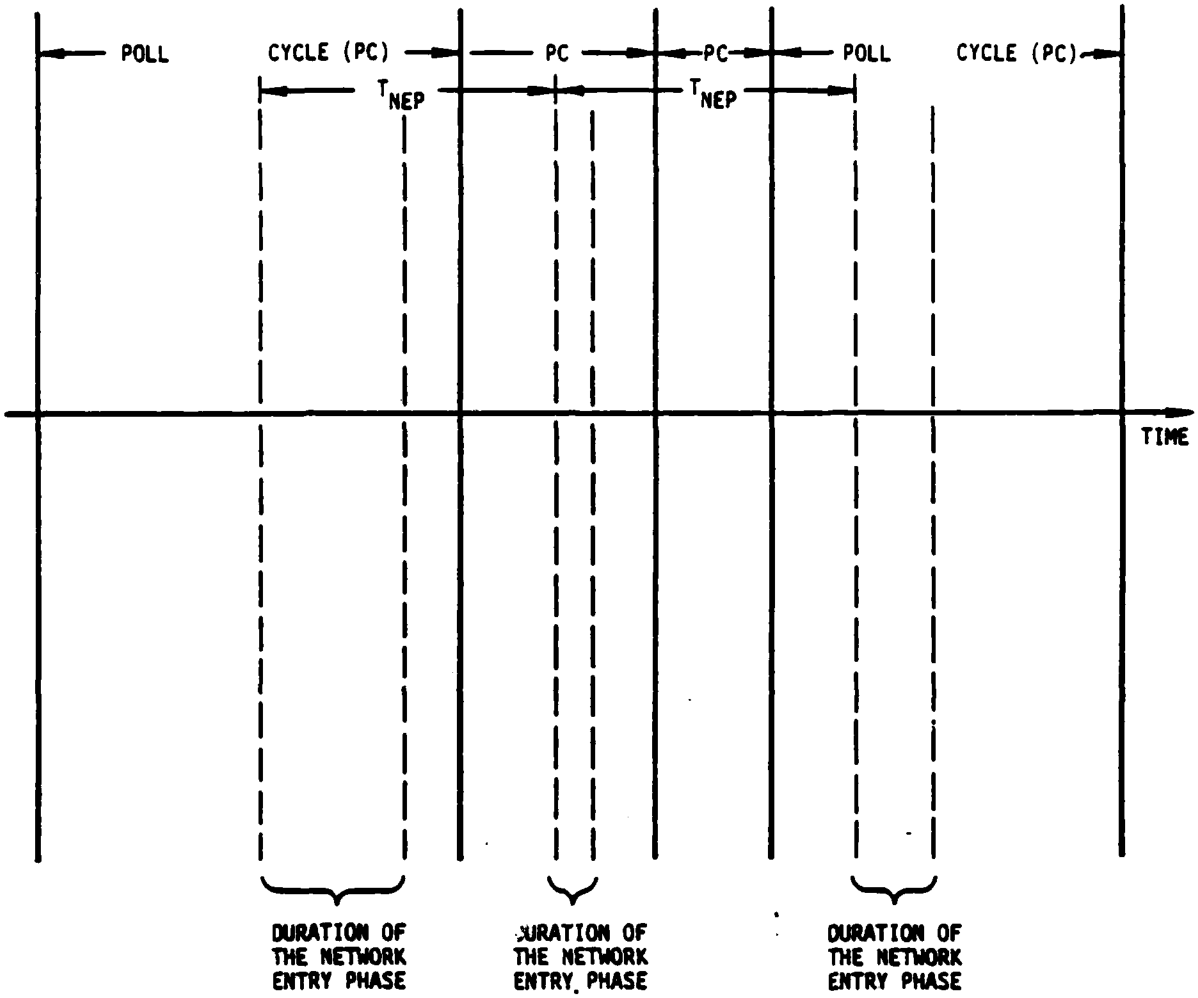

POLL CYCLE (PC)
PC
PC
POLL CYCLE (PC)
T_NEP
T_NEP
TIME
DURATION OF THE NETWORK ENTRY PHASE
DURATION OF THE NETWORK ENTRY PHASE
DURATION OF THE NETWORK ENTRY PHASE

FIGURE 3. RELATIONSHIP BETWEEN THE POLL CYCLE, THE DATA EXCHANGE PHASE AND THE NETWORK ENTRY PHASE

NCS and the ship and are therefore transparent to the access and link control protocols described here. Consider that the NCS has two messages: Message A (to ship 1) and Message B (to ship 2). Message A has a higher priority than Message B. The NCS will transmit the lower priority Message B before Message A if the polling order requires ship 2 to be polled before ship 1. This disadvantage of the protocol should be studied further. Incorporating priorities into the link and delivering frames in order appears complex or intractable. The natural solution to the problem is to employ two logical links corresponding to the two priorities. Then the NCS transmission algorithm would give corresponding higher priority in selecting the links to be serviced.

The ships inform the NCS of their queue sizes. Based on this information, the NCS determines the amount of traffic the ships are allowed to transmit to the NCS. This is described in Section 6 of this volume.

This description has assumed the existence of a network with the NCS as the primary station. However, it is possible to have multiple NCS. This results in a network with several primary stations and many secondary stations. Each primary station serves its own contingent of ships. The selection of a primary station by a ship is done by the operator in the ship. Since the primary stations may all use the same group of frequencies for transmission, collisions will sometimes occur between the transmissions of the different primary stations. Also, since the primary stations may use the same group of frequencies for reception of frames from the ships, collisions will sometimes occur between the transmissions from the different ships. This problem is not severe, since the different NCS are located at great distances from each other.

3. FRAMES

Transmissions between higher level protocols at the NCS and the ships occur in entities called messages. As illustrated in Figure 4, messages are partitioned into segments. Thus the transmission of a message is carried out by the transmission of segments, which includes higher level protocol information (such as flow control, network entry/exit, etc.). This combined entity is referred to as a block. Finally, a block is encapsulated in a frame for the link level transmission between the NCS and the ships.

Every new transmission to a receiver must begin with a preamble. This preamble is used to synchronize the sender and the receiver. The preamble carries such information as, for example, the code rate, encryption/decryption message indicator etc. The duration of the preamble is estimated to be 1.6 seconds for both modem and crypto synchronization.

As every new transmission must precede with a preamble, it is advantageous to combine transmissions. A transmission comprising of a preamble followed by several concatenated frames is more efficient than transmitting the same frames in different groups with each group needing a preamble.

The frame is the primary unit of transmission at the link level and as shown in Figure 4, is comprised of the following fields:

- Flag
- Destination Address
- Source Address
- Control
- Block
- Frame Check Sequence (FCS).

Each of these fields is defined and their bit lengths estimated in the following section.

3.1 Definition of Fields

3.1.1 Flag

Two flags, the begining flag and the ending flag (Figure 4), enclose the frame. The beginning flag serves as a reference for the position of the destination address, source address and the control fields and initiates transmission error checking; the ending flag

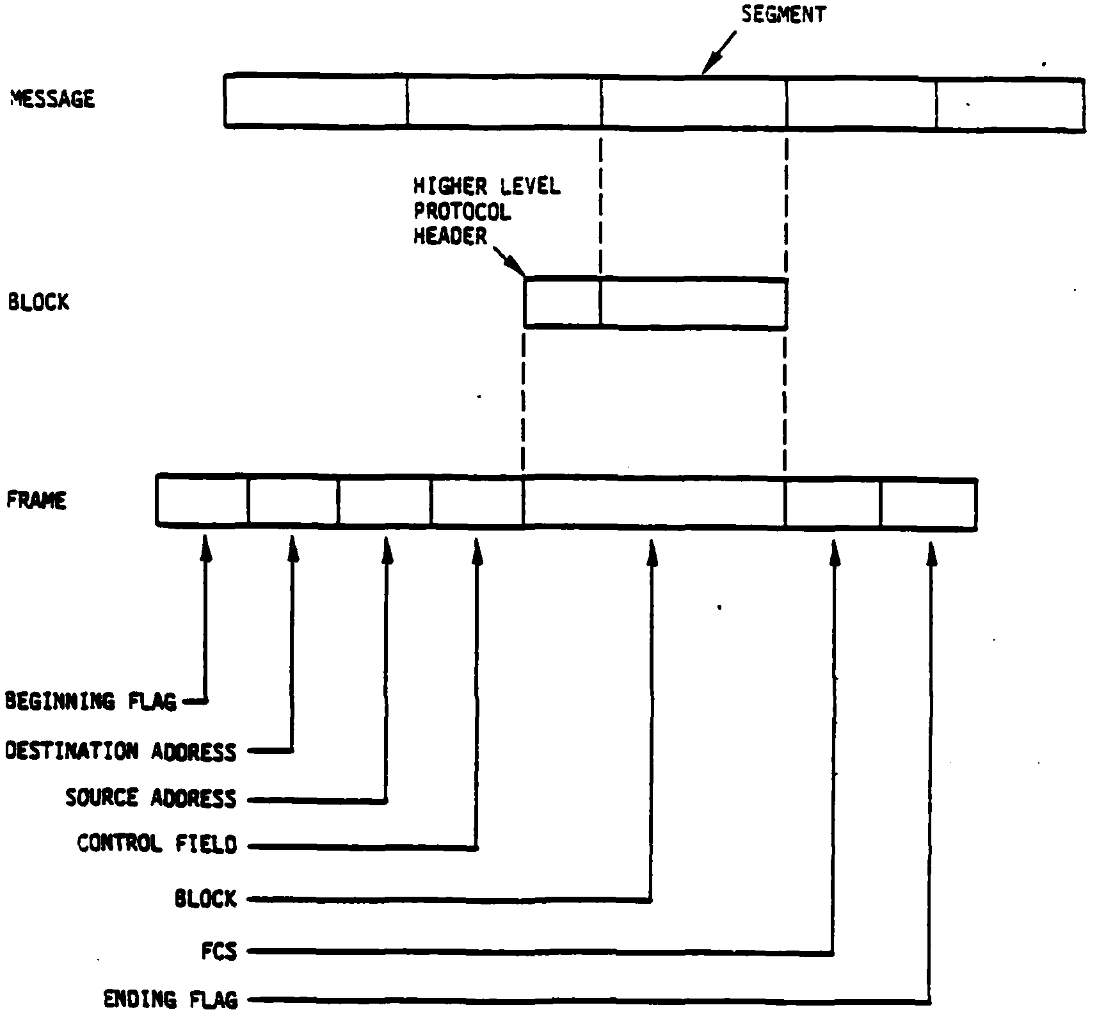


FIGURE 4. REPRESENTATION OF TRANSMISSION ENTITIES

terminates the check for transmission errors. Both beginning and ending flags are 8 bits long and the binary configuration 01111110 is recommended. The bit orientation of the protocol allows the flags to be recognized at any time. A flag may be followed by a frame, by another flag or by an idle condition.

3.1.2 Zero Insertion

A frame is identifiable because it begins with a flag and contains only non flag bit patterns. (The frame ends at the next flag). This characteristics does not restrict the contents of a frame because a binary 0 must be inserted by the transmitter after any succession of five contiguous 1s within the frame. Thus, no pattern of 01111110 is ever transmitted by chance. After testing for flag recognition, the receiver removes a 0 that follows a received succession of five contiguous 1s. Inserted and removed 0s are not included in the transmission error check.

3.1.3 Destination Address

The destination address identifies the receiver of the frame. Frames directed to the ships are addressed by the ship sequence number, which is assigned to the ships during the Network Entry Phase. In addition, a broadcast address is defined such that all ships recognize this address as their own. Finally, frames directed to the NCS are addressed by the NCS ID. The destination address field has a length of 8 bits.

3.1.4 Source Address

The source address identifies the sender of the frame. Frames sent by the ships contain the ship sequence number as the Source Address. Whereas, frames sent by the NCS contain the NCS ID as the source address. The source address field has a length of 8 bits.

3.1.5 Control

The control field contains the capability for encoding commands and responses required to control a data link. The contents of this field are described in detail when individual frames are introduced in Section 3.2. The length of the control field varies from 4 bits to 33 bits, depending on the type of frame.

3.1.6 Block (Information)

The block contains data that is moved, via the data link, between the NCS and the ships. The block field is unrestricted in format or content; its contents are transparent to the components of data link control. A block field is normally included with every frame having a control field of the I-frame format. These I-frames are the only ones that are sequenced. The block field has a maximum length of 1024 bits. A possible enhancement to the protocol is to dynamically set the maximum length block at net entry or more frequently.

3.1.7 FCS (Frames Check Sequence)

The FCS field, with recommended field size of 16 binary digits, follows the block field (if there is one; the control field, if not) and immediately precedes the ending flag. These 16 digits result from a mathematical computation, known as cyclic redundancy checking, on the digital value of all binary bits (excluding inserted 0s) within the frame; the purpose is to validate transmission accuracy.

The transmitter performs the computation and sends the resulting FCS value. The receiver performs a similar computation and checks its results. The receiver discards a frame that is found to be in error. The specific polynomial to be employed in computing the CRC is an implementation issue.

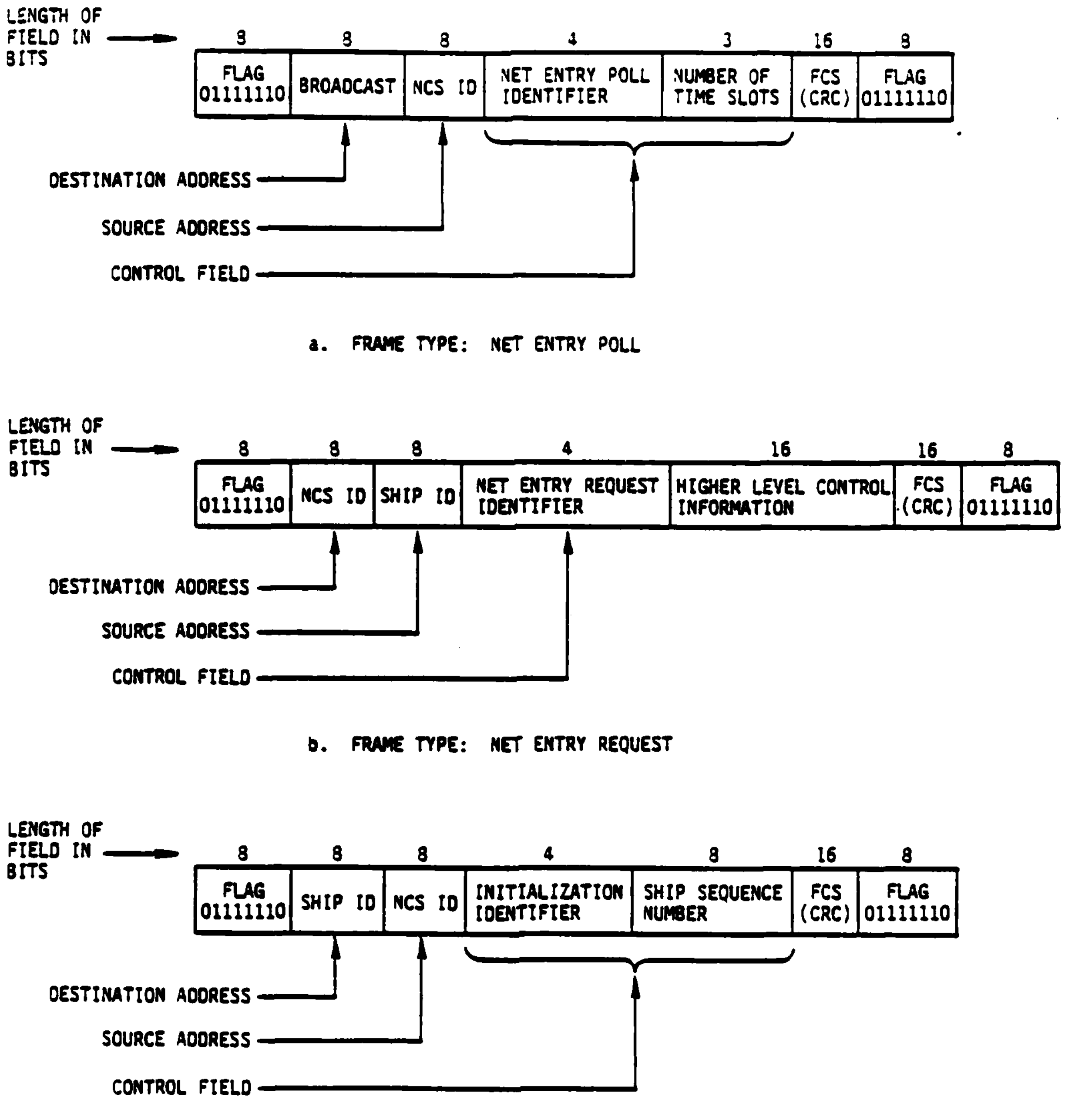
3.2 Frame Types

The frames that are transmitted between the NCS and the ships, during the Network Entry Phase, are shown in Figure 5. The frames that are transmitted by the NCS to the ships, during the Data Exchange Phase, are shown in Figure 6. The frames that are transmitted by the ships to the NCS, during the Data Exchange Phase, are shown in Figure 7.

The detailed definition of the fields in all of the frames, identified in Figures 5, 6, and 7, is given in the context of their use in Sections 4, 6 and 7.

3.3 Frame Handling

All frames transmitted by a sender and received by a receiver undergo a sequence of processing operations. A frame, that is to be transmitted by the sender, will have its CRC

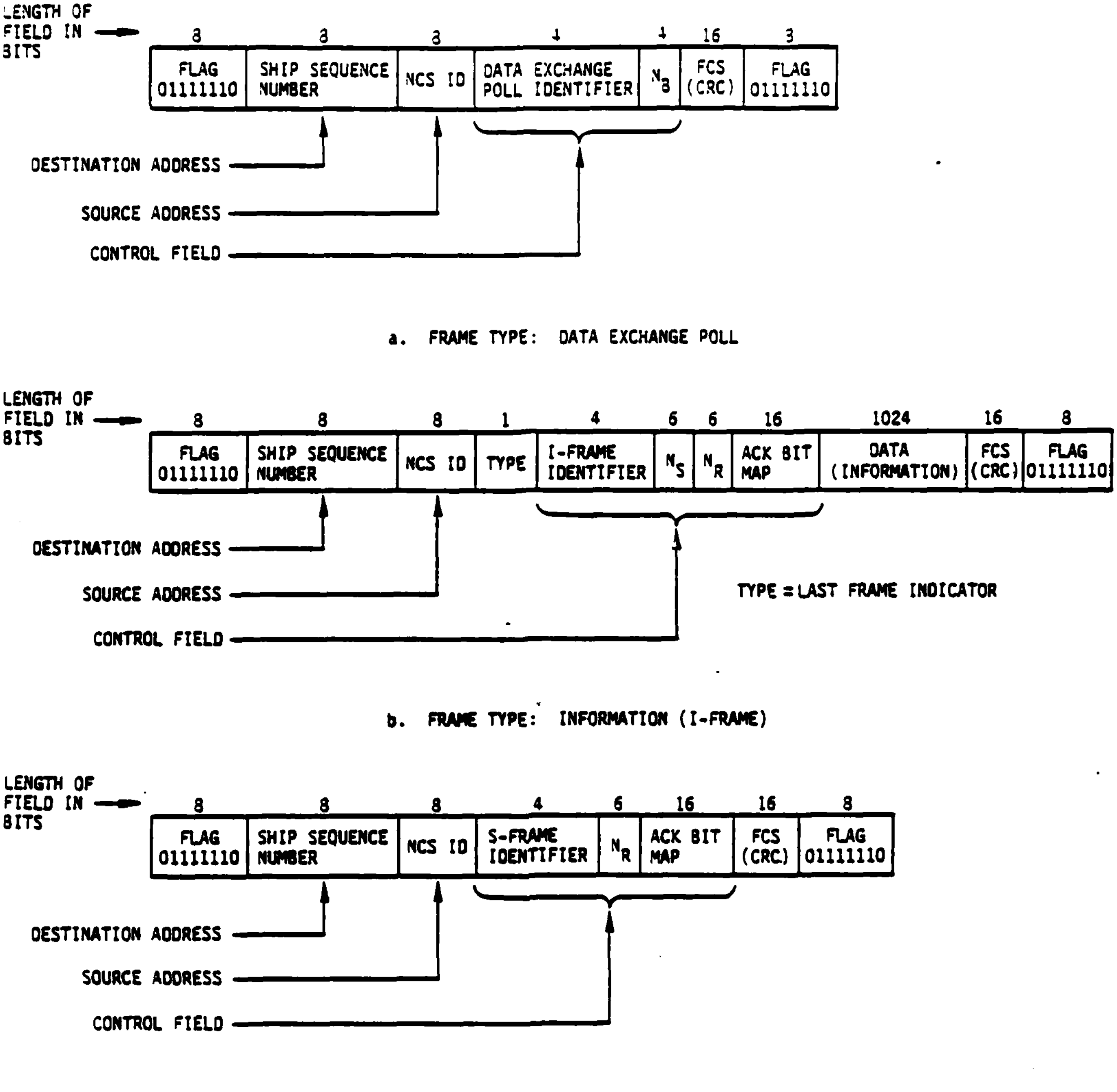


a. FRAME TYPE: NET ENTRY POLL

b. FRAME TYPE: NET ENTRY REQUEST

c. FRAME TYPE: INITIALIZATION

FIGURE 5. FRAMES USED IN THE NETWORK ENTRY PHASE

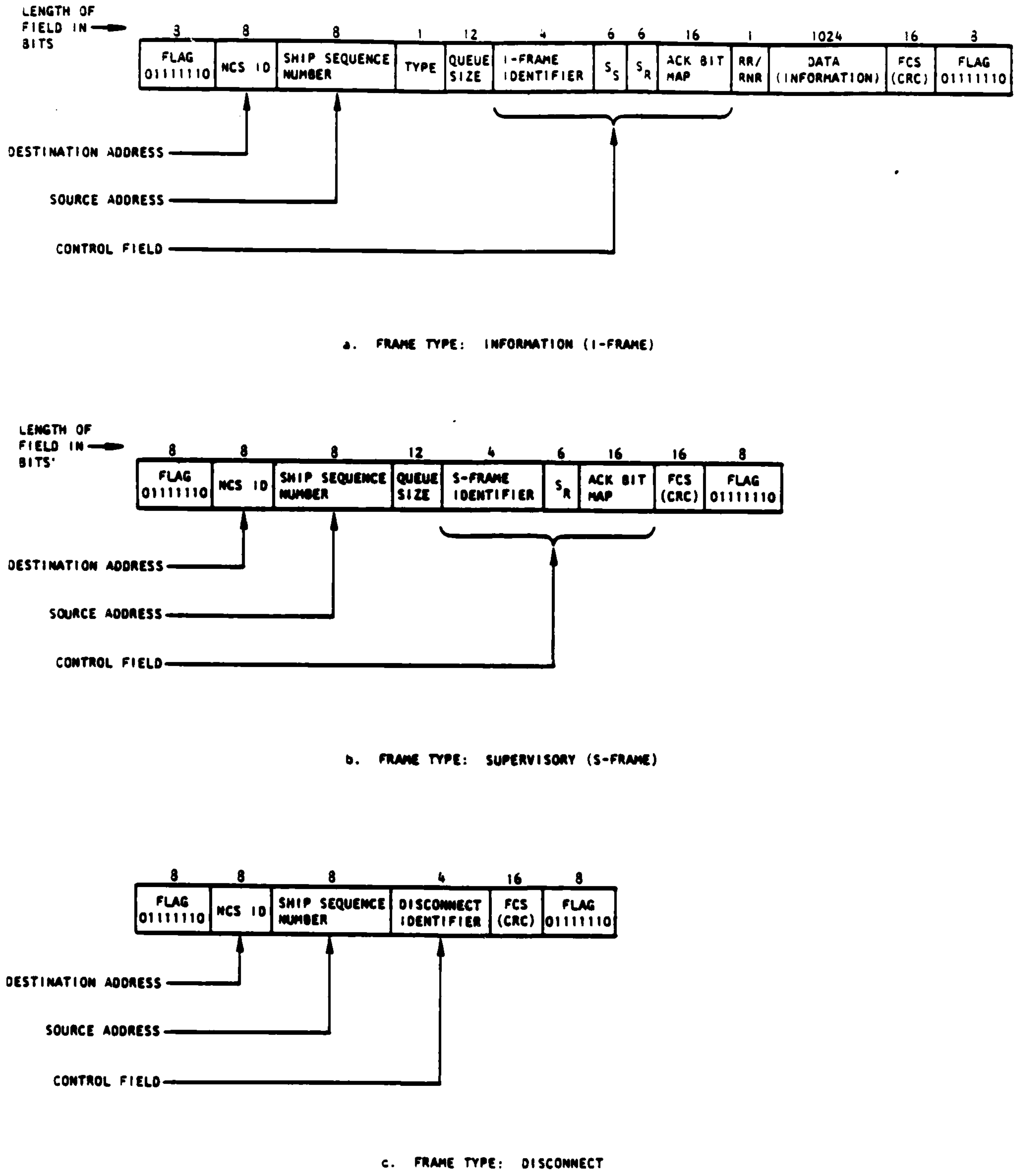


a. FRAME TYPE: DATA EXCHANGE POLL

b. FRAME TYPE: INFORMATION (I-FRAME)

c. FRAME TYPE: SUPERVISORY (S-FRAME)

FIGURE 6. FRAMES TRANSMITTED BY THE NCS DURING THE DATA EXCHANGE PHASE



a. FRAME TYPE: INFORMATION (I-FRAME)

b. FRAME TYPE: SUPERVISORY (S-FRAME)

c. FRAME TYPE: DISCONNECT

FIGURE 7. FRAMES TRANSMITTED BY THE SHIPS DURING THE DATA EXCHANGE PHASE

computed. This CRC is inserted in the FCS of the frame. The frame is provided with data transparency by the process of zero insertion. Beginning and Ending flags are added to the frame. The frame, is, then, handed over to the Encryption device where it is encrypted. Following encryption, the frame is coded by the modem. If this is the first frame in a transmission, a preamble is appended to the front of the frame. The frame is, then, transmitted over the radio channel.

A receiver continuously monitors the receive radio channels listening for transmissions from the sender. A received frame is decoded by the modem. The decryption device is used to decrypt the frame. The frame undergoes a reverse zero insertion to remove the zeros, inserted by the sender to ensure data transparency. CRC on the frame starts after detection of the beginning flag. This CRC is compared with the FCS, in the last field, of the frame before the end flag. The frame is discarded if the comparison fails. On the other hand, if the computed CRC matches the FCS in the frame, the beginning and ending flags are removed from the frame. The transmitted frames are received by all receivers. But only the receiver, for whom the frame is addressed, should receive it. Thus, the frame is discarded by the receiver if the destination address in the frame does not match the address of the receiver.

4. NETWORK ENTRY PHASE

4.1 Introduction

The Network Entry Phase is the period of time during which ships are granted the opportunity to enter the network. This phase is initiated at fixed time intervals (a parameter, T_{NEP}, shown in Figure 3, having, for example, a value of 2 minutes). Operating asynchronously relative to the poll cycle, it may be initiated at any instant within the poll cycle. However, the phase is constrained to start after completion of transmission of any frame to a ship.

The particular protocol that will be used for the Network Entry Phase is referred to as Modified Aloha with Expanding Slots. This protocol will be described in the remainder of this section.

4.2 Modified Aloha With Expanding Slots

For the purpose of this protocol, the NCS serves as the master while all the ships serve as slaves. A diagram illustrating the frames transmitted between the NCS and the ships (during the Network Entry Phase) is given in Figure 8.

The NCS initiates the Network Entry Phase by broadcasting a Net Entry Poll to ships within the network and to ships outside the network. This poll defines the time slots that are available for the ships outside the network to transmit their access requests to the NCS. The time slots begin after the ships receive the Net Entry Poll (as shown in Figure 8).

Ships which desire to enter the network will transmit their Net Entry Request in one of the time slots. Collisions occur whenever two or more ships transmit their Net Entry Requests during the same time slot.

The NCS will transmit an Initialization frame to every ship (whose Net Entry Request did not collide with the Net Entry Requests from other ships) accepted into the network. If collisions occurred between two or more Net Entry Requests the NCS will retransmit the Net Entry Poll, with a larger number of time slots. (This capability is based on the assumption that collisions can be detected.)

The functional flow diagram, for the process described above, is illustrated in Figure 9. The dotted line indicates retransmission of the Net Entry Poll in case of collisions between Net Entry Requests. Some of the functions identified in Figure 9 occur at the NCS

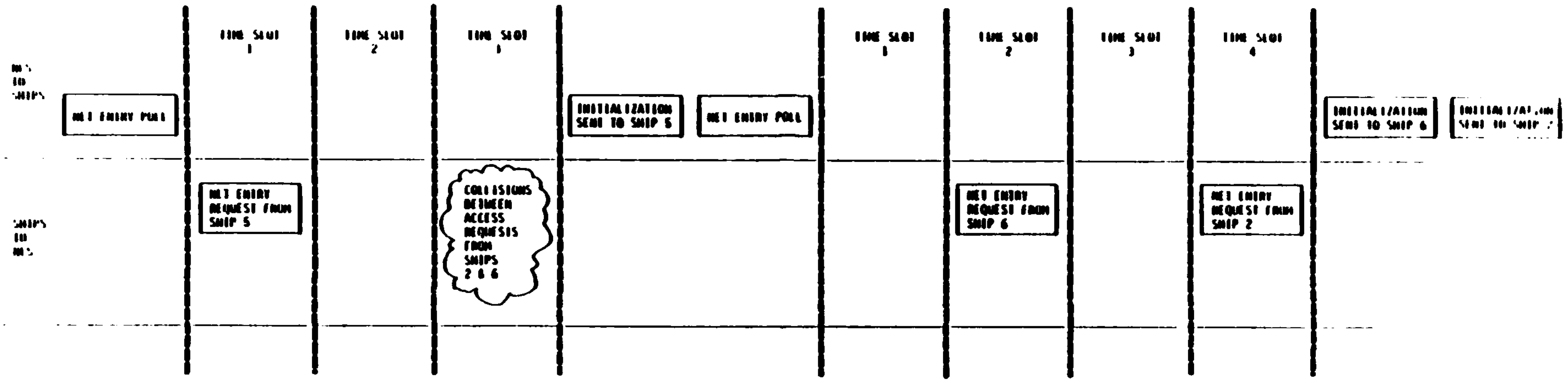

NCS TO SHIPS
SHIPS TO NCS
NET ENTRY POLL
TIME SLOT 1
NET ENTRY REQUEST FROM SHIP 5
TIME SLOT 2
TIME SLOT 1
COLLISIONS BETWEEN ACCESS REQUESTS FROM SHIPS 2 & 6
INITIALIZATION SENT TO SHIP 5
NET ENTRY POLL
TIME SLOT 1
TIME SLOT 2
NET ENTRY REQUEST FROM SHIP 6
TIME SLOT 3
TIME SLOT 4
NET ENTRY REQUEST FROM SHIP 2
INITIALIZATION SENT TO SHIP 6
INITIALIZATION SENT TO SHIP 2

FIGURE 8. DIAGRAM ILLUSTRATICNG MODIFIED ALOHA WITH EXPANDING SLOTS

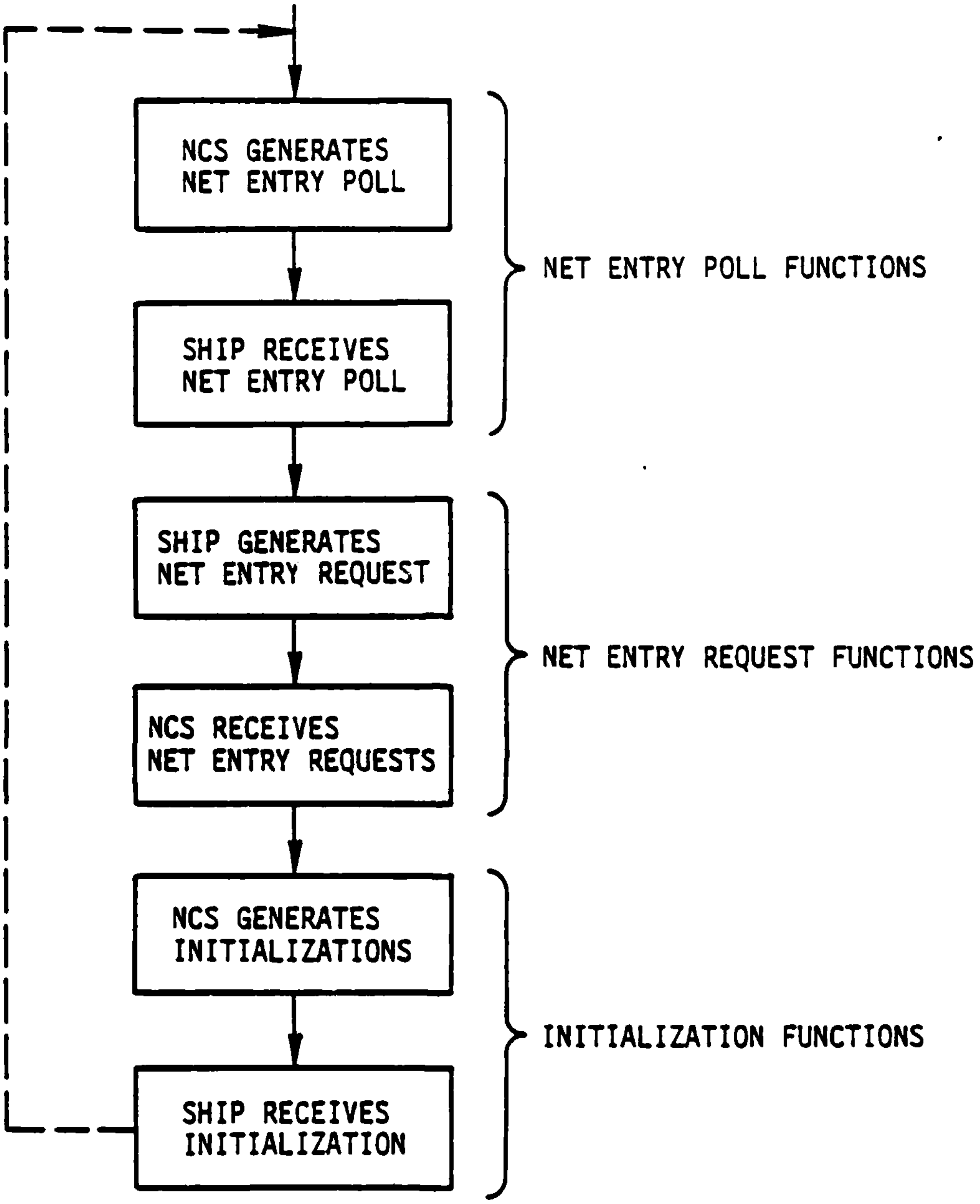

NCS GENERATES NET ENTRY POLL
SHIP RECEIVES NET ENTRY POLL
NET ENTRY POLL FUNCTIONS
SHIP GENERATES NET ENTRY REQUEST
NCS RECEIVES NET ENTRY REQUESTS
NET ENTRY REQUEST FUNCTIONS
NCS GENERATES INITIALIZATIONS
SHIP RECEIVES INITIALIZATION
INITIALIZATION FUNCTIONS

FIGURE 9. FUNCTIONAL FLOW DIAGRAM FOR MIDIFIED ALOHA WITH EXPANDING SLOTS

while others occur at the ships. Each of these functions is described in the following sections.

4.3 Net Entry Poll

The Net Entry Poll is broadcast by the NCS to all ships within and outside of the network. The format of the Net Entry Poll frame has been illustrated in Figure 5(a). The frame defines the number of time slots available for a ship to transmit a Net Entry Request.

The functions associated with Net Entry Poll are divided between the NCS (which broadcasts the Net Entry Poll) and the ships (which receive the Net Entry Poll). The Net Entry Poll functions at the NCS (Net Entry Poll Generation) will be first described followed by a description of the Net Entry Poll functions at the ship (Net Entry Poll Reception).

4.3.1 Net Entry Poll Generation

The Network Entry Phase is initiated at fixed time intervals (for example, two minutes). At these times, the Data Exchange Phase is informed that the Network Entry Phase is waiting to begin. In the full duplex mode of operation the NCS will complete transmission of the frame in progress and then transmit the Net Entry Poll. While waiting for the responses to the Net Entry Poll, the NCS resumes sending data to the ships that have already been initialized. However such transmissions will be at the low information rate of 300 bps.

The NCS generates a Net Entry Poll using an Initial Value for the number of time slots. This Initial Value is a parameter with a typical value of two. The Net Entry Poll is broadcast to all ships. The NCS starts a timer (see Initialization Generation) when the Net Entry Poll is broadcast.

The Net Entry Poll is also regenerated and rebroadcasted, whenever collisions occurred between Net Entry Requests. This is described in Section 4.5.1.

4.3.2 Net Entry Poll Reception

All ships tuned in to the receive frequencies will receive the Net Entry Poll, if the CRC on the frame is successful. If the ship is not trying to enter the network, the Net Entry Poll is discarded and the ship returns to the listen mode (waiting for frames from the

NCS). If the ship is trying to enter the network, it proceeds to initiate access requests (Net Entry Request Generation).

4.4 Net Entry Request

A ship desiring to enter the network will transmit a Net Entry Request to the NCS. The format of the Net Entry Request frame has been illustrated in Figure 5(b).

The functions associated with Net Entry Request are divided between the ships (which transmit the Net Entry Request) and the NCS (which receives the Net Entry Request). The Net Entry Request functions at the ship (Net Entry Request Generation) will be first described, followed by a description of the Net Entry Request functions at the NCS (Net Entry Request Reception).

4.4.1 Net Entry Request Generation

A ship, trying to enter the network, will generate a Net Entry Request on receipt of a Net Entry Poll from the NCS. The Net Entry Request will contain, as one of the items in the Higher Level Control Information field, a value for the size of queue (of blocks to be sent to the NCS). It will, next, determine which NCS provided time slot to use for transmitting its Net Entry Request to the NCS.

Let s = number of time slots specified by the Net Entry Poll
for example, s ranges in value between 2 and 8 inclusively.

Then the probability of a ship transmitting its Net Entry Request in one of these s time slots is given by

$$\text{Probability} = \frac{1}{s}.$$

The ship uses a randomizing process to select one of the s time slots.

The ship begins counting time slots after receipt of the Net Entry Poll, using the SYSGEN parameter, time slot duration. The ship transmits the Net Entry Request to the NCS during its selected time slot. After transmission, the ship returns to the listen mode (waiting for frames from the NCS).

4.4.2 Net Entry Request Reception

After the broadcast of the Net Entry Poll, the NCS listens for Net Entry Requests from the ships. The NCS will listen for Net Entry Requests until the end of all time slots. Any Net Entry Requests received after expiration of all time slots are discarded by the NCS and the NCS returns to the listen mode (waiting for frames from ships).

All non-colliding Net Entry Requests, which arrive at the NCS, before the end of the last time slot, are processed by Initialization Generation (Section 4.5.1). Colliding Net Entry Requests are discarded by the NCS and the occurrence of collision is passed, for use by Initialization Generation.

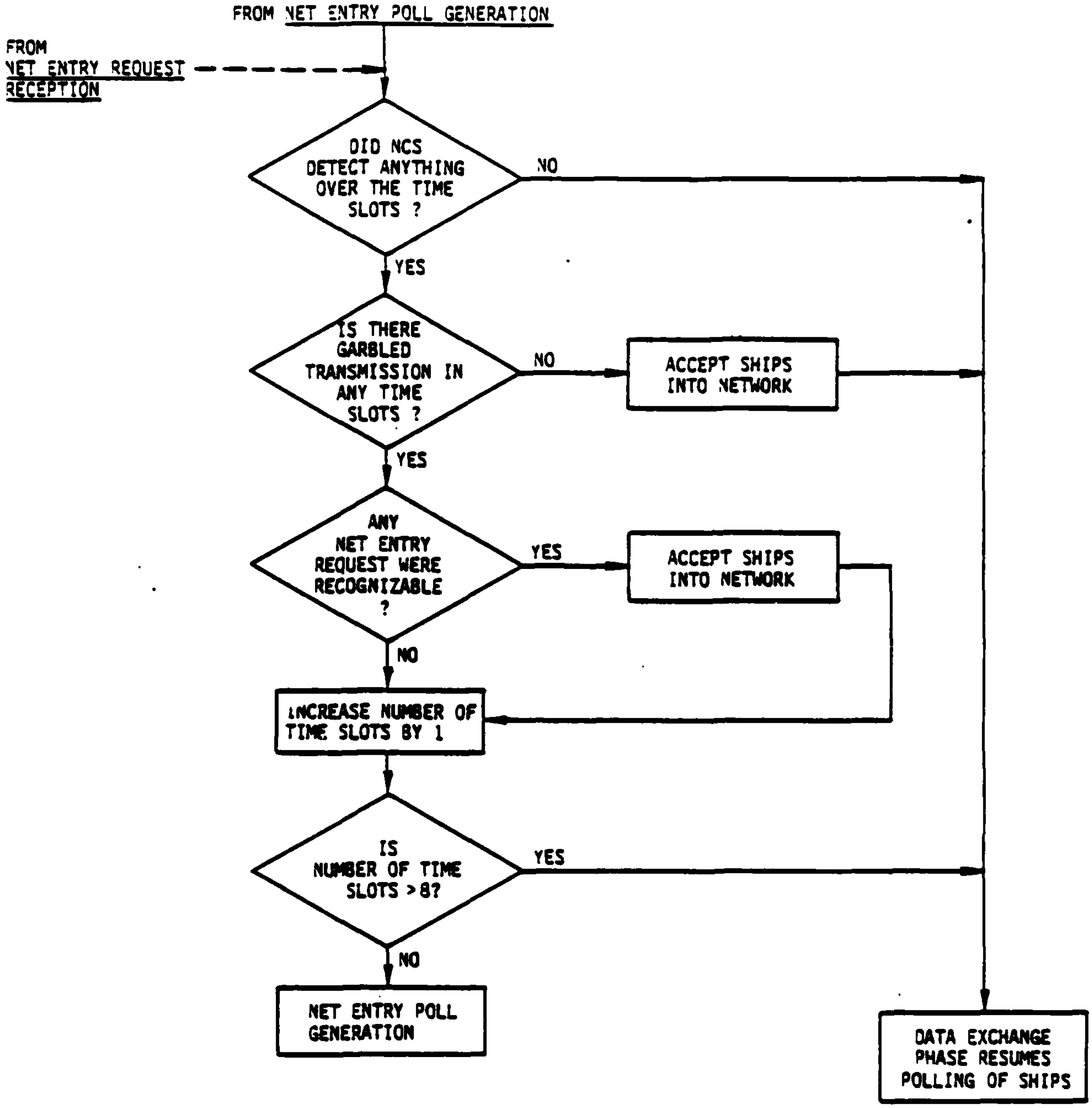
4.5 Initializaton

The Net Entry Requests recognized by the NCS, identified the ships requesting entry into the network. These ships are granted permission to enter the network, whenever the NCS sends Initialization frames to the ships, with each ship receiving its own Initialization frame. The format of the Initialization frame has been illustrated in Figure 5(c). The frame shows a ship sequence number which is used by the NCS to identify the ship within the network. The ship sequence number also serves as a confirmation to the ship, of its entry into the network.

The functions associated with Initialization are divided between the NCS (which transmits the Initialization frames) and the ships (which receive the Initialization frames). The Initialization functions at the NCS (Initialization Generation) will be first described, followed by a description of the Initialization functions at the ships (Initialization Reception).

4.5.1 Initialization Generation

The functions at the NCS (Initialization Generation) are illustrated by the flow diagram in Figure 10. The details illustrated in this figure are described below. If nothing was received by the NCS, the Network Entry Phase is terminated and the Data Exchange Phase resumes at the point it was suspended. If the NCS received only non-colliding Net Entry Requests, the NCS will accept these ships into the network. (See the flow diagram in Figure 11).


FROM NET ENTRY POLL GENERATION
FROM NET ENTRY REQUEST RECEPTION
DID NCS DETECT ANYTHING OVER THE TIME SLOTS ?
NO
YES
IS THERE GARBLED TRANSMISSION IN ANY TIME SLOTS ?
NO
ACCEPT SHIPS INTO NETWORK
YES
ANY NET ENTRY REQUEST WERE RECOGNIZABLE ?
YES
ACCEPT SHIPS INTO NETWORK
NO
INCREASE NUMBER OF TIME SLOTS BY 1
IS NUMBER OF TIME SLOTS >8?
YES
NO
NET ENTRY POLL GENERATION
DATA EXCHANGE PHASE RESUMES POLLING OF SHIPS

NOTE: THE DOTTED LINE INDICATES THAT NET ENTRY REQUESTS (STORED BY NET ENTRY REQUEST RECEPTION) WILL BE USED HERE. NET ENTRY REQUEST RECEPTION DOES NOT INITIATE INITIALIZATION GENERATION

FIGURE 10. INITIALIZATION GENERATION AT NCS

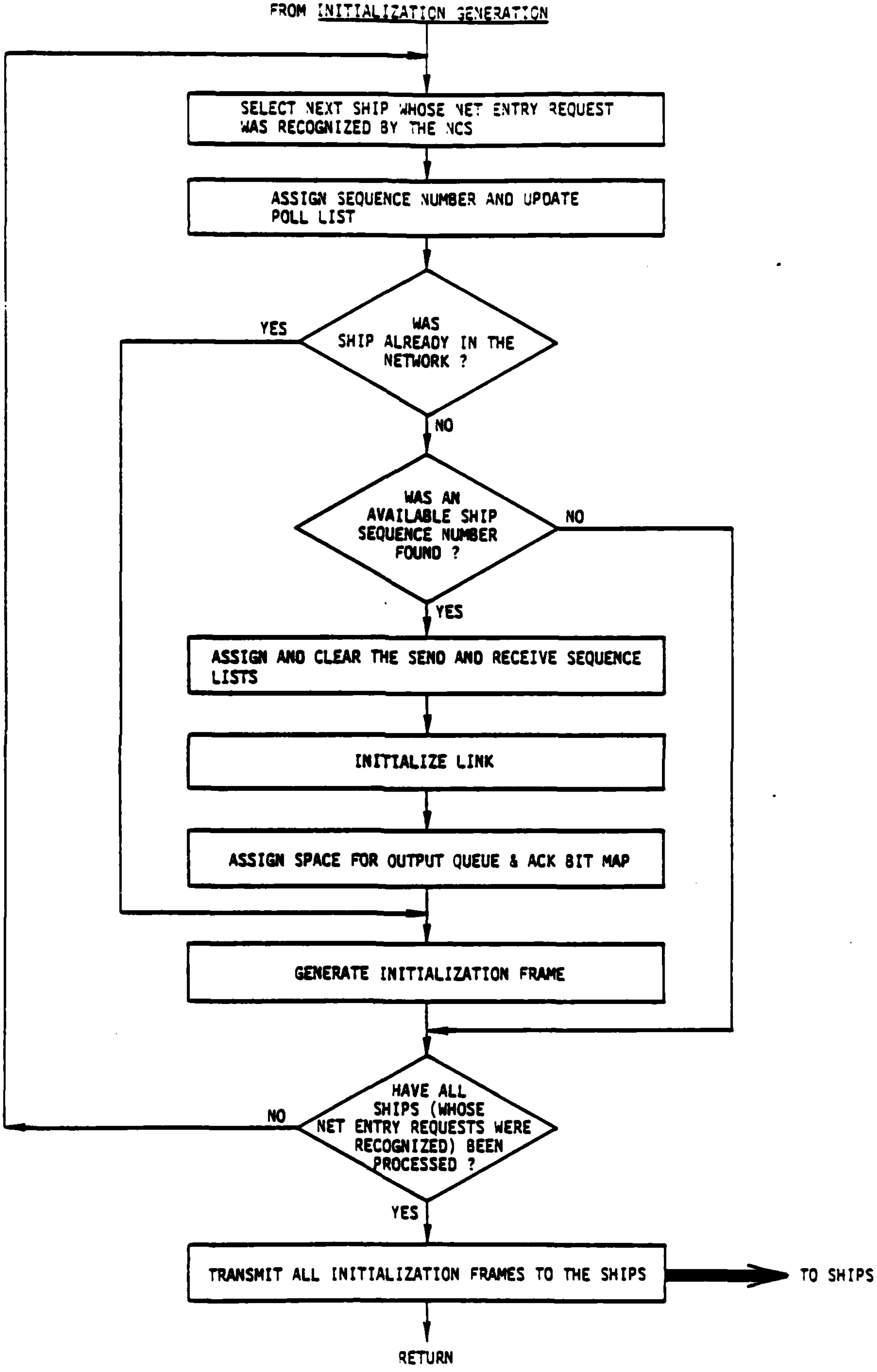

FROM INITIALIZATION GENERATION
SELECT NEXT SHIP WHOSE NET ENTRY REQUEST WAS RECOGNIZED BY THE NCS
ASSIGN SEQUENCE NUMBER AND UPDATE POLL LIST
WAS SHIP ALREADY IN THE NETWORK ?
YES
NO
WAS AN AVAILABLE SHIP SEQUENCE NUMBER FOUND ?
NO
YES
ASSIGN AND CLEAR THE SEND AND RECEIVE SEQUENCE LISTS
INITIALIZE LINK
ASSIGN SPACE FOR OUTPUT QUEUE & ACK BIT MAP
GENERATE INITIALIZATION FRAME
HAVE ALL SHIPS (WHOSE NET ENTRY REQUESTS WERE RECOGNIZED) BEEN PROCESSED ?
NO
YES
TRANSMIT ALL INITIALIZATION FRAMES TO THE SHIPS
TO SHIPS
RETURN

FIGURE 11. ACCEPT SHIPS INTO NETWORK

Each ship, accepted into the network, will be assigned a ship sequence number. This number is unique and is assigned only at network entry. The ship sequence number is used by the NCS to identify a ship in its network and its receipt by a ship also serves as a confirmation to the ship of its entry into the network. The assignment of the ship sequence numbers is illustrated by the flow diagram in Figure 12. Before assigning a ship sequence number to a ship (requesting entry into the network), the NCS checks for the ship's presence in the Poll List.

The Poll List (illustrated in Figure 13) contains an entry for every ship accepted into the network and contains the ship ID, the code rate for transmissions with the ship and the ship sequence number assigned to the ship. The ships in the Poll List are grouped together according to the code rate at which they communicate with the NCS. As seen in Figure 13, the group of ships at top of the Poll List form the first group and the ships in this group communicate at the lowest code rate. The group of ships at the bottom of the Poll List form the last group and the ships in this group communicate at the highest code rate. The polling cycle begins with the first group of ships and ends with the last group of ships. It is noticed in Figure 13, that all the ships have unique ship sequence numbers. These ship sequence numbers range from one to the maximum numer of ships allowed in the network.

If the ship already exist in the Poll List, the NCS retrieves the previously assigned ship sequence number. Such a case occurs whenever the NCS receives a Net Entry Request from a ship already in the network. This may be due to the ship failing to receive an Initialization frame from the NCS after the ship's previous attempt at Net Entry, where the NCS received the ships Net Entry Request and sent the ship an Initialization frame. Here the NCS had allowed the ship to enter the network, but the ship was unaware of the fact and so attempted another network entry.

If the ship does not already exist in the Poll List, the NCS assigns the ship a sequence number. For example, one alternative is to scan the assigned ship sequence numbers in the Poll List and locate the numerically lowest unassigned ship sequence number. This number can be assigned to the ship. An entry is made for the ship within the group of ships (in the Poll List) communicating at the same code rate as the requesting ship. This procedure ensures a repetitive use of the ship sequence numbers without having duplicate assignment.

In case a ship sequence number is not available (i.e., the network contains the maximum allowable number of ships), the ship is not accepted into the network. The NCS does not perform Initialization for the ship and no Initializaton frame is sent to the ship. This is expected to occur rarely and would be handled manually by having the operator retry later or to attempt communication with another NCS.

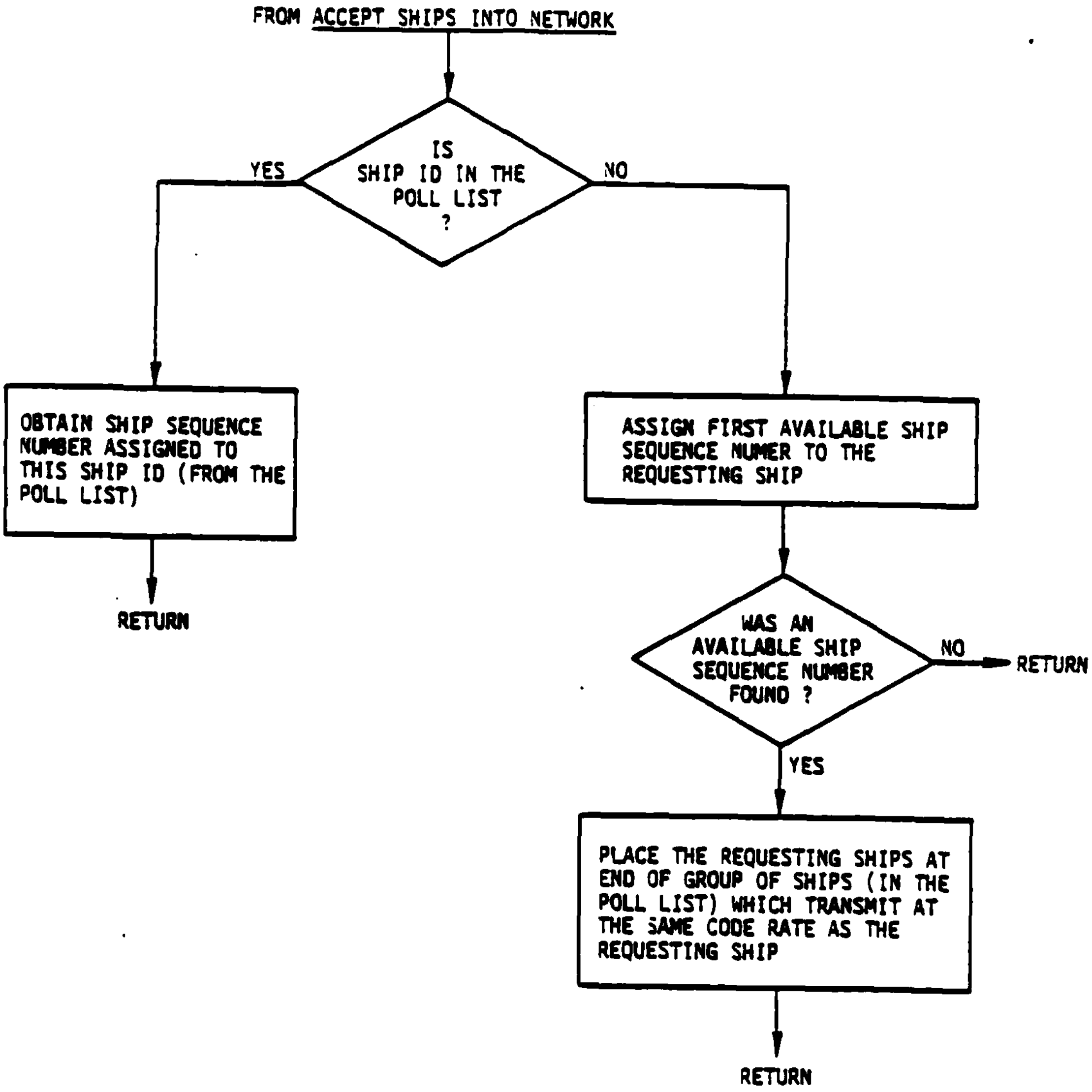

FROM ACCEPT SHIPS INTO NETWORK
IS SHIP ID IN THE POLL LIST ?
YES
NO
OBTAIN SHIP SEQUENCE NUMBER ASSIGNED TO THIS SHIP ID (FROM THE POLL LIST)
RETURN
ASSIGN FIRST AVAILABLE SHIP SEQUENCE NUMER TO THE REQUESTING SHIP
WAS AN AVAILABLE SHIP SEQUENCE NUMBER FOUND ?
NO
RETURN
YES
PLACE THE REQUESTING SHIPS AT END OF GROUP OF SHIPS (IN THE POLL LIST) WHICH TRANSMIT AT THE SAME CODE RATE AS THE REQUESTING SHIP
RETURN

FIGURE 12. ASSIGN SEQUENCE NUMBER AND UPDATE POLL LIST

POLL LIST

SHIP SEQUENCE NUMBER	SHIP ID	CODE RATE
1	X1	300
3	AB	300
100	X5	300
20	CD	300
98	X3	300
4	X2	600
5	YY	600
6	BC	600
	⋮	⋮
50	X10	2400
52	MM	2400

FIGURE 13. POLL LIST STORED AT THE NCS

As seen in Figure 11, for every ship which is accepted for the first time into the network, the link protocol must be initialized. This includes allocation and clearing of send and receive sequence lists. The send and receive windows are initialized at beginning of the respective sequence lists. In addition, space for the Output Queue and ACK bit map is allocated at this time. An Initialization frame is generated for the ship.

The NCS performs the above for every ship whose Net Entry Request was recognized. At end, the NCS successively transmits each of the generated Initialization frames to the respective ships. At this point the Network Entry Phase is terminated and the full duplex Data Exchange Phase resumes at the point it was suspended.

Referring to Figure 10, it is seen that it is assumed to be possible for the NCS to recognize garbled frames and collisions. Net Entry Requests, which were distinguishable from the collision, are treated the same way as described previously. However, after transmission of the Initialization frames to the ships, accepted into the network, the NCS rebroadcasts the Net Entry Poll with a higher value for the number of time slots.

Finally, if all the transmissions received by the NCS, during all the time slots, were garbled, the NCS rebroadcasts the Net Entry Poll with a larger value for the number of time slots. The NCS keeps rebroadcasting the Net Entry Poll with successively higher values for the number of time slots until a Net Entry Poll with a recommended value of eight time slots is broadcast.

If garbled transmission is still received by the NCS, (probably the result of excessive channel noise), the Network Entry Phase is terminated and the Data Exchange Phase resumes at the point it was suspended. The exception condition is noted to the NCS operator or higher level protocol.

4.5.2 Initialization Reception

An Initialization frame addressed to a ship will be received by the ship if the frame passes CRC. If the ship is not trying to enter the network, the Initialization frame is discarded. The Initialization frame is, again, discarded if the ship is not waiting for a response to a previously transmitted Net Entry Request. However, if the ship is waiting for the Initialization frame, receipt of the frame informs the ship of its entry into the network. The ship, on knowing of its entry into the network, will initialize the link.

A ship, waiting for an Initializaton message, may, instead, receive another Net Entry Poll or a Data Exchange Poll or it may timeout for lack of response, from the NCS, to a transmitted Net Entry Request. Either of these cases is an indication to the ship that either the Initialization sent by the NCS to the ship was lost or the NCS did not receive the ship's previously transmitted Net Entry Request (due to either collisions with other Net Entry Requests or too much noise on the channels). A ship will, in such a case, attempt Network Entry, again, when it receives another Net Entry Poll. The number of times a ship will attempt Network Entry, before giving up, is a SYSGEN parameter. After this number of attempts, a ship may proceed with failure recovery actions.

5. DATA EXCHANGE PHASE

As described in Section 2, full duplex communications between the NCS and the ships occurs during the Data Exchange Phase. Communications with the ships is controlled by the NCS, serving as a master. Ships are granted permission, to have full duplex communication with the NCS, by means of a Data Exchange Poll. As described in Section 2, ships are grouped by their code rates so that ships in a group may be polled in succession without the need for code rate synchronization. Code rate synchronization is necessary whenever transmission shifts from one code rate to another. It is always possible to have transmission at a lower code rate with ships rated at higher code rates, but the opposite is not true. Thus the NCS transmission algorithm (described in Section 6) requires polling to proceed from low code rates to high code rates. Within a group of ships (with the same code rate) the NCS polls the ships round robin in the order dictated by their relative position in the Poll List. It is recommended to poll ships only once during a poll cycle (instead of some ships with high volume multiple times) because of the high overhead associated with synchronization.

Once a ship is granted access to the NCS, full duplex communication can occur with the NCS. Figure 14 presents an overview of this process. Detailed explanation of some of the terminology, used in this figure, is postponed to the next two sections. It is seen that the top half of Figure 14 is almost duplicated in the bottom half but in the reverse direction. This is due to the two way communication of the data link between the NCS and a ship.

High level processes (such an Application Processes and Net Exit) wishing to send messages to the receiver, will queue their message blocks at the Output Queue. The messages in the Output Queue are ordered by their priorities. The high priority messages are ahead of the low priority messages. Also, within a priority class, the messages are ordered First Come First Serve (FCFS) i.e., a message that arrives at the Output Queue before another one will be placed in front of the later arriving messages, as long as both messages have the same priority.

The network layer at the sender will queue blocks at the link level. The link level will select blocks at the head of this queue. The link level will merge ACKs contained in the ACK bit map with the block in the send window before the access level transmits the I-frames to the receiver. The blocks are accountable and the sender needs a copy of the block in case the block needs to be retransmitted. However the ACKs are not accountable and thus the sender need not keep a copy of them. Thus, the control field of a retransmitted I-frame will not be identical to its previously transmitted version because the ACKs which

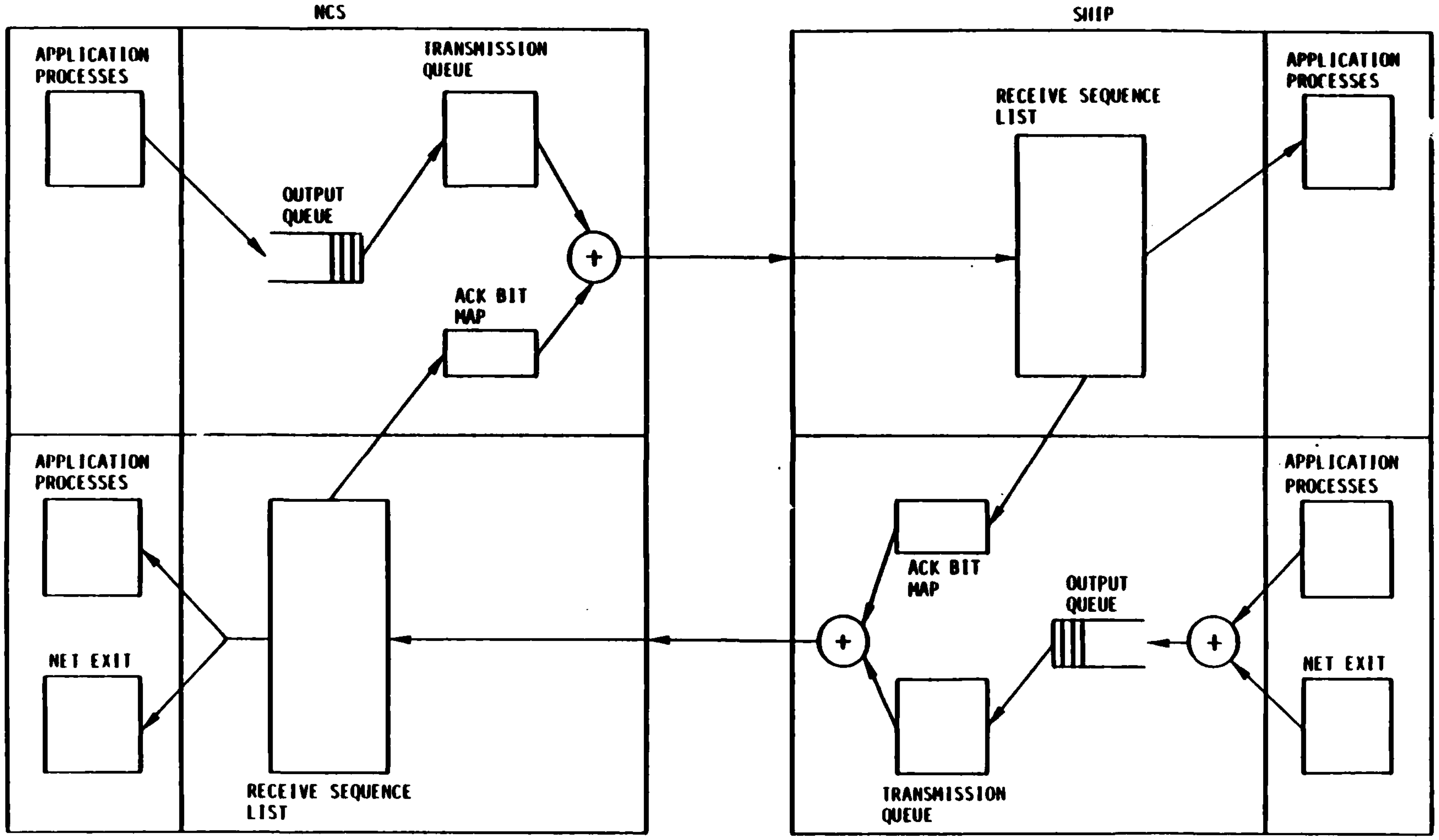


FIGURE 14. FULL DUPLEX COMMUNICATION BETWEEN THE NCS AND A SHIP

the I-frames piggyback will be different (the state of the ACK bit map being different, at the different times the I-frame is transmitted and retransmitted).

The link level at the receiver holds the I-frames in the receive window. The link level ACKs these correctly received blocks by setting the bits in the ACK bit map where a bit, in the ACK bit map, represents the corresponding location of the block in the receive window. The link level forwards all blocks to the higher layer processes at the receiver in sequential order for message reassembly.

The above discussion is valid for transmissions from a NCS to a ship and vice versa. Section 6 will discuss the Access Level while Section 7 will deal with the link level.

6. ACCESS LEVEL FOR DATA EXCHANGE

This section describes the access level protocol used by the NCS and the ships.

6.1 Polling Of A Ship

The access level, at the NCS, will initiate the Data Exchange Phase at the next ship in the Poll List. The ship, selected to be polled, is examined to see if it is in the radio silent mode. If the selected ship is in the normal mode, a Data Exchange Poll is sent to the ship. The poll informs a ship of the maximum number of frames, N_B, it may send to the NCS when it is polled. The determination of this quantity is described in Section 6.6. The format of the Data Exchange Poll has been illustrated in Figure 6(a). A response timer is started when the poll is sent to a ship. Its purpose is to detect a lack of response by the ship to the poll.

If the ship is in the radio silent mode, the access level, at the NCS does not send a Data Exchange Poll to the ship, but instead sends frames addressed to the ship according to the NCS transmission algorithm. Ships in the radio silent mode are assumed capable of receiving at an information rate of only 300 bps. In the radio silent mode, the NCS cannot expect any response from the ship.

6.2 Receipt of Frames At A Ship

6.2.1 Receipt of Data Exchange Poll/Ship Transmission

The ships enter the Data Exchange Phase on receipt of a Data Exchange Poll. The access level, at the ships, will communicate with the link level to obtain the frames that must be sent. Whether I-frames or S-frames are being sent is transparent to the link. The access level will set the frame counter to one after sending the first frame to the NCS and thereafter increase the frame counter by one for each frame sent to the NCS.

The access level will set the last frame indicator bit in the frame if either the frame counter reaches its maximum value, N_B, or the link level signals that a frame it is passing to the access level is the last frame.

6.2.2 Receipt Of Other Frames

The access level at the ship will pass all frames received from the NCS in the Data Exchange Phase (other than poll frames), to the link level of the ship.

6.3 Expiration of Timeouts

6.3.1 No Response To Data Exchange Poll

As mentioned in Section 6.1, a response timer was started by the access level, at the NCS, after transmission of the Data Exchange Poll. If no response was received from the ship, the response timer would expire. This is an indication that either the ship failed to receive the Data Exchange Poll or that its response to the Data Exchange Poll was unrecognizable to the access level, at the NCS.

In either case, the access level, at the NCS, retransmits the Data Exchange Poll to the same ship. The number of times a Data Exchange Poll is retransmitted, to the same ship, is a parameter. When this number of retransmissions is reached, the access level, at the NCS, assumes that the ship is inactive. The access level, at the NCS, then proceeds to poll the next ship in the Poll List.

6.3.2 Interframe Timeout Expiration

If the access level, at the NCS, receives a response from the ship to a Data Exchange Poll, it will start an interframe timer. The timer is set to timeout at the end of the ship's transmission period. A ship's transmission period is the time required for the ship to send N_B frames to the NCS. If the interframe timeout expires before the NCS has received a frame with the last frame bit set (e.g., the last frame may have been lost due to transmission error), then, the access level, at the NCS, will proceed to poll the next ship in the poll list.

6.4 Receipt of Frames at NCS

The access level, at the NCS, will pass the link and higher level protocol fields of the frames, received from the ships to the link level. The access level examines the last frame indicator bit in the frame received from the ship. If the last frame indicator bit is set, the access level, at the NCS, proceeds to poll the next ship in the poll list.

6.5 NCS Transmission Algorithm

The ships serve as secondary stations and, as such, can transmit to the NCS only when the NCS sends them Data Exchange Polls. On the other hand, since the NCS is the master, it can transmit to the ships at any time. Thus it is allowed for the NCS to transmit to a ship even though the ship is not currently polled. This method results in better use of the transmission capacity because if the NCS has nothing to send to the polled ship, instead of remaining idle, the NCS can send any available frames to the nonpolled ships.

The access level, at the NCS, monitors its transmission need to the currently polled ship. This transmission requirement has the highest priority. If the NCS has nothing to send to the currently polled ship the access level will search for frames, to transmit to non polled ships, in the same group as the currently polled ship. This search for frames, within the group, will proceed round robin from one ship to the next in the same order as the order of the ships in the Poll List. If the access level finds frames to be sent, to a ship in the group, it will send at most one frame to the ship; then it will resume its search with the next ship in the group. The other frames will be sent to the ship during successive search cycles through the group, one frame sent per search cycle. The frames are sent to the nonpolled ships at the same code rate as the currently polled ship.

The search within a group proceeds until as many frames as can be sent to ships (for whom the access level found frames to be transmitted) have been sent. At this time the access level begins to search the next group in the Poll List. The access level transmits frames to this new group of nonpolled ships the same way as described for the previous group. The search proceeds from one group in the Poll List to the next in round robin. This search is limited to groups which have code rates higher than or equal to the code rate of the currently polled or to be polled ship. Transmissions with the new group of ships will occur at the same code rate as the currently polled ship.

As mentioned previously, NCS transmissions to the currently polled ship have the highest priority. If the access level finds that it has something to send to the currently polled ship, it will interrupt the low priority process of sending frames to the nonpolled ships and will honor the transmission request to the currently polled ship. After honoring this high priority request, the access level will resume with the low priority process at the point where it was interrupted.

6.6 Capacity Allocation Algorithm

Each ship, in the network, is allocated a certain amount of time during which it may, when polled by the NCS, exchange information with the NCS. This is equivalent to the ship being allocated transmission capacity. An algorithm, for allocating the transmission capacity, is presented in the following section.

6.6.1 Algorithm

In this section, an algorithm for capacity allocation based on load is described. Let

$N_i(1)$ = number of priority 1 blocks in queue at ship i.
$N_i(2)$ = number of priority 2 blocks in queue at ship i.
a = weighting factor of relative importance of priority 1 to priority 2.

Then define the equivalent load for ship i as

$$N_i(e) = aN_i(1) + N_i(2).$$

Transmission duration time of ship i, T_i, is then allocated proportionally

$$T_i = T \cdot N_i(e) / \sum N_i(e)$$

where

T = duration of the poll cycle.

To determine the number of blocks that can be transmitted, let

L = frame length
S = transmission speed (dependent on code rate)
B = transmission time of one frame.

Then

$$B = L/S$$

And the number of blocks that can be transmitted by the ship i in time T_i is given by

$$N_{B_i} = T_i/B.$$

6.6.2 Implementation

The capacity allocation algorithm, executed by the access level in the NCS, requires that each of the ships, in the network, inform the NCS of their respective $N_i(1)$ and $N_i(2)$ values. This information is sent to the NCS:

- At the time the ships enter the network.
- During the time the ships are in the Data Exchange Phase (i.e., the ships are polled by the NCS).

At the time they enter the network, the ships inform the NCS access level of the $N_i(1)$ and $N_i(2)$ values, by including these quantities in the Net Entry Request (this was covered in Section 4.4.1). During the time they are in the Data Exchange Phase, the ships inform the NCS access level of the $N_i(1)$ and $N_i(2)$ values, by having their access levels include these quantities in the first frame sent to the NCS (see Figure 7(a), 7(b)). The queue size field, in the I and S frames, is a continuation field in that the frame contains an indicator bit which is set if the field is present and not set if the queue size field is absent. Thus, the first frame sent by a ship, in response to a Data Exchange Poll, will have the indicator bit set implying the presence of the queue size field. Any sucessive frames sent by the ship will have the indicator bit not set.

There are two modes in which the ships collect their queue size information: synchronous and asynchronous. In the synchronous mode, the ship's access level obtains new $N_i(1)$ and $N_i(2)$ values every poll cycle. The values obtained during the current poll cycle represent the ship's projected needs during the next poll cycle. The capacity allocation algorithm (at the NCS) is executed every poll cycle, after all the ships have been polled. The new capacity allocations take effect during tne next poll cycle.

In the asynchronous mode, the ship's access level obtains new $N_i(1)$ and $N_i(2)$ values only when these values exceed threshold parameters. Each ship possesses its own upper and lower thresholds for the size of the queue. If the actual number of blocks in the queue either falls below the lower threshold or rises above the upper threshold, the access level collects these new $N_i(1)$ and $N_i(2)$ values. The capacity allocation algorithm (at the NCS) is executed every poll cycle, after all the ships have been polled. However, the capacity requirements are a combination of new ones generated by the ship's access levels during the current poll cycle and old ones generated by the ship's access levels during previous poll cycles. The new capacity allocations takes effect during the next poll cycle.

7. LINK LEVEL PROTOCOL

7.1 Overview

This section describes the link level protocol used by the NCS and the ships to communicate with each other. The link level, at the sender will pass frames to the access level which, in turn, sends the frame to the receiver. The link level, at the receiver will receive frames passed to it by the access level.

Section 7.2 gives the definitions of terms used in the link level protocol. Section 7.3 illustrates the use of the protocol. Section 7.4 describes the initialization procedure. Section 7.5 describes the send procedure while Section 7.6 describes the receive procedure. Section 7.7 describes the retransmission procedure. Section 7.8 pertains to the Network Exit function while Section 7.9 describes the Application Interface.

7.2 Definitions

Figure 15 depicts an example of the following definitions.

7.2.1 Sequence Space

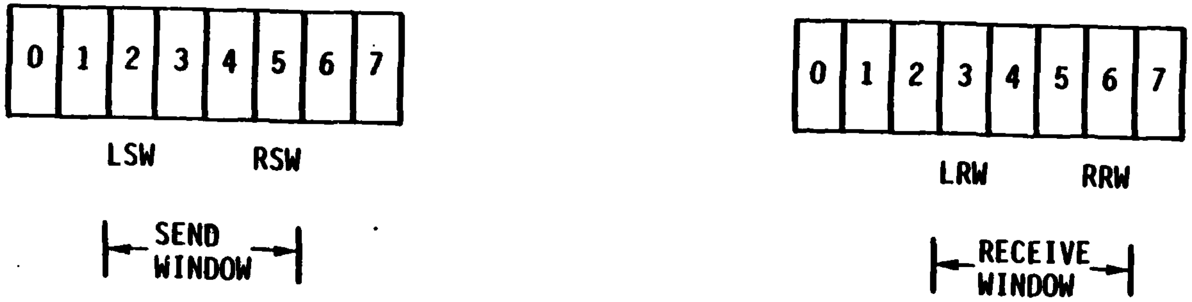
The sequence space is defined to be a set of integers (0, 1, 2, ..., 2^{k-1}). The size of the sequence space, cardinality of S is 2^k, where k is an integer.

7.2.2 Window Size

The window size, W, is defined as the maximum number of outstanding unacknowledged frames. Thus the sequence number of a frame to be transmitted can never be more than W greater than the last frame acknowledged. The worst case situation occurs when the receive window is completely ahead of the send window. Thus, the receiver can expect retransmitted I-frames (which are duplicates) one window behind its current receive window (the previous receive window).

The need for duplicate reception by the receiver implies that the receiver needs 2W unique I-frame numbers. Thus the cardinality of the sequence space S must satisfy:

S greater than 2W.



FRAME 2 TRANMITTED BY SENDER

FRAME 2 RECEIVED BUT NOT ACKNOWLEDGED

FIGURE 15. WINDOW EXAMPLE

When S is less than 2W, the receiver cannot distinguish new I-frames from duplicate I-frames in its previous receive window, and a resulting ambiguity occurs.

7.2.3 Send State

The send process in the link protocol can be defined in terms of the following variables. The Left Send Window, LSW, corresponds to a frame in the send sequence list which has the following properties:

1. All frames prior to it have been acknowledged by the receiver.

2. The frame itself has yet to be ackowledged by the receiver.

The Right Send Window, RSW, corresponds to a frame in the send sequence list which has the sequence number

$$RSW = \quad LSW+W-1.$$

Then the send window is comprised of all frames between the right window and the left window, inclusively. Only frames in the send window are eligible for transmission. These frames may be in the following status:

1. Waiting to be sent to the receiver; these frames are in the link transmission queue.

2. Sent to the receiver, but are awaiting acknowledgement from the receiver; these frames are in the link retransmission queue.

3. Sent to the receiver and have been acknowledged by the receiver; these frames can be discarded.

The link level, at the sender, passes frames located in the send window, to the access level. These frames may be either retransmitted frames or first time frames. In either case, $N_S(S_S)$ represents the sequence number of the frame, in the send window, which is passed by the link level to the access level.

7.2.4 Receive State

Analogously the receive process in the link protocol can be defined. Accountable frames (e.g., message blocks) are held by the receiver in a receive sequence list. This allows the link level to pass the frames in sequence to the network level. The Left Receive Window, LRW, corresponds to a frame in the receive sequence list which has the following properties:

1. All frames prior to it have been received correctly; the receiver has set the ACK bits in the ACK bit map for these frames and these frames have been forwarded to the network level.

2. The frame itself has yet to be received from the sender.

The N_R in the I-frame and S-frame is set equal to the LRW.

The Right Receive Window, RRW, corresponds to a frame in the receive sequence list which has the sequence number

$$RRW = \quad LRW+W-1.$$

Then the receive window is comprised of all frames between the right window and the left window, inclusively. These frames may be in the following states:

1. Not yet been received by the receiver.

2. Received by the receiver but the ACK bits in the ACK bit map have not been set.

3. Received by the receiver and the ACK bits in the ACK bit map have been set.

A receiver ACKs a correctly received I-frame by setting a bit in the ACK bit map. The location of this bit, in the ACK bit map, matches the location of the I-frame in the receive window. A bit, in the ACK bit map, which is not set represents the fact that the receiver has not correctly received the corresponding I-frame, in the receive window. Since the receive window is not static, but moves within the Receive Sequence List, it is necessary to give the bits in the ACK bit map a relative address to match with the current location of the receive window. This is done by the use of the quantity LRW which gives the first bit in the ACK bit map an address equal to the first I-frame in the receive window.

7.3 Example of Selective Repeat Procedure

7.3.1 Overview

The Link Control Operation follows the selective repeat procedure. Let the maximum number of outstanding unacknowledged I-frames at the sender be W, and the size of the sequence space be S. Then the window size (which is the maximum allowable difference in sequence number between unacknowledged I-frames) for this selective repeat scheme is S -W. With these figures, the size of the buffer required at the sender is W times the average I-frame size.

7.3.2 Walk Through of the Selective Repeat Procedure

The Selective Repeat Procedure is illustrated in this section. As shown in Figure 16, the send/receive sequence spaces have a size of 7. The windows in either of the sequence lists are outlined by the dark verticle lines.

The sender is in a state where the send window contains I-frames 3, 4 and 5 with SN_S=3. The receiver is in a state where the receive window waits for I-frames 3, 4 and 5. This is represented in Figure 16 by Condition 1.

The sender transmits I-frames 3, 4 and 5 to the receiver. The receiver receives I-frame 4 and 5 but does not receive I-frame 3. The state at the sender and receiver at this point is given by Condition 2.

The receiver ACKs I-frames 4 and 5. The sender receives these ACKs and Condition 3 represents the state of the sender and receiver.

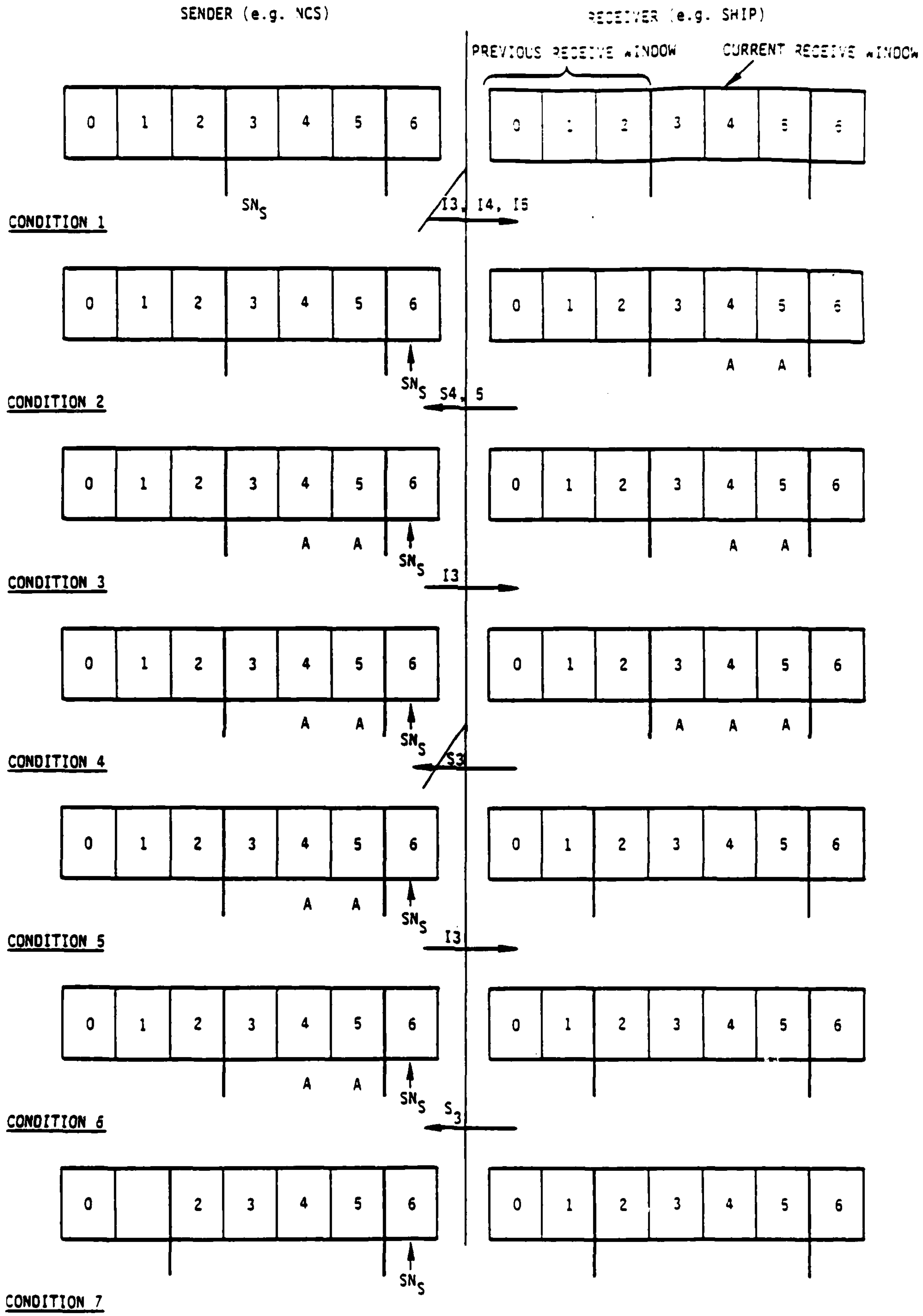


FIGURE 16. OPERATION OF THE WINDOW MECHANISM AT SENDER AND RECEIVER

Since the sender has not received an ACK for I-frame 3, within a predefined time interval, the sender retransmits I-frame 3. This time, the receiver correctly receives the retransmitted I-frame 3. Condition 4 represents the state of the sender and receiver at this point in time.

The receiver ACKs I-frame 3 and advances the receive window. However, the sender fails to receive the ACK and thus cannot advance its send window. The state of the sender and receiver at this point is shown by Condition 5.

Again, the sender retransmits I-frame 3 at the end of a predefined period of time. The receiver interprets this I-frame as a duplicate and sends an ACK to the sender. The states at the sender and the receiver remain unchanged, as shown by Condition 6.

When the sender receives the ACK for I-frame 3, it advances its send window. The state of the sender and receiver at this point is shown by Condition 7.

From the above description, it is seen that the send window will either always lag or keep up with the receive window.

The receive window will advance forward only when the first I-frame in the receive window has been ACKed. When this happens, the receive window will advance forward until the first location in the receive window is empty. The I-frames leaving the receive window are forwarded to the higher level processes.

The send window will advance forward only when the first I-frame in the send window has received an ACK. When this happens, the send window will advance forward until an un-ACKed I-frame appears in the first location of the send window.

The sender can transmit I-frames, in its send window, in any order it chooses to. Retransmission of an I-frame does not require retransmissions of ACK'ed I-frames which are ahead of it, in the send window. After, all I-frames, in the send window, ha' e been transmitted, new I-frames in the send sequence list cannot be transmitted unless the send window is advanced forward.

7.4 Initialization

To initialize the link processes, set

LRW = 0 LSW = 0,
RRW = W RSW = W.

Enter at most W frames into the link transmission queue at the access level.

7.5 Send

7.5.1 Overview

According to the access level protocol, when it is time for a transmission, the access level will trigger the link level requesting a frame for transmission. There are two cases to consider as described below:

- The link level has a I-frame for transmission.
- The link level has no I-frames for transmission.

7.5.2 No I-frames for Transmission

In the former case there are two subcases:

- Always send a S-frame.
- Send a S-frame only if a received I-frame is to be ACKed.

The former option would be implemented in the ship and the latter option in the NCS. In both cases at the ship, the link level would indicate to the access level that there are no more frames for transmission so the access level can set the last frame bit.

The format of the S-frames are depicted in Figures 6(c) and 7(b).

7.5.3 I-frames Ready for Transmission

When an I-frame is queued for transmisison, the link level will:

- Choose the I-frame with the smallest send sequence number (modulo S) currently in the transmission queue.

- Set a time-out for retransmission of this frame.

- Increment number of transmissions for this frame.

- Put this frame into the retransmission queue.

- Form a piggyback acknowledgement.

- Pass I-frame for transmission to access level.

The format of the I-frames are depicted in Figure 6(b) and 7(a).

7.5.4 Radio Silent Receiver

A ship in the radio silent mode can only receive transmission from the NCS. The ship cannot transmit to the NCS. This means that the NCS can send S-frames to the radio silent ship without difficulty. However, the NCS can send I-frames to the ship only under the condition that their transmission is not held up by lack of ACKs arriving from the ship. In fact, in the radio silent mode, the ship will not ACK I-frames from the NCS and the NCS will transmit I-frames to the ship as fast as it is able to. The concept of a send window at the NCS (and a receive window at the ship) is no longer necessary.

A block in the send window is selected. Then, any ACKs, in the ACK bit map, are piggybacked onto the I-frame containing this block. This I-frame is then sent to the ship.

If the send window has more I-frames to send to the ship, the window is advanced. Blocks which have been sent from the send window are replaced by additional blocks from the network layer, and the process described above is repeated.

The transmission of I-frames to the radio silent ship continues until the last I-frame has been sent to the ship.

7.6 Receive

7.6.1 I-Frames and S-Frames

The link level receives I-frames from the access level. These I-frames are used to update the receive window and the piggybacked ACKs are used to update the send window. The link level receives S-frames from the access level. These S-frames contain ACKs and are used to update the send window. The link level will reset the response timer on receipt of either I or S frames.

7.6.2 Update Receive Window

(A) Previous Window

An I-frame received for a location which was in the previous receive window is a duplicate. Such a duplicate arrives because the sender failed to receive the ACK sent by the receiver to the initial I-frame. In response the link level will send an S-frame (or piggybacked I-frame) with the current ACK bit map to the sender. This is an implicit confirmation to the sender that the receiver has received all I-frames prior to the left receive window. The link level discards the duplicate I-frame.

(B) Current Window Previously ACKed

An I-frame received for a location in the current receive window which is already ACKed, is also a duplicate. Again, such a duplicate arrives because the sender failed to receive the ACK sent, by the receiver, to the initial I-frame. The response of the link level will be to send an S-frame (or piggybacked I-frame) with the current ACk bit map to the sender and discard the duplicate I-frame.

(C) Current Window Not Previously ACKed

An I-frame received for a location in the current receive window which has not been ACKed, is an original I-frame. The link level sets the ACK bit, in the ACK bit map (representing the current receive window) and, if possible, proceeds to advance the receive window. If all preceding frames have been received, the link will forward all frames to the network layer (that can be forwarded in sequence).

(D) Beyond Current Window

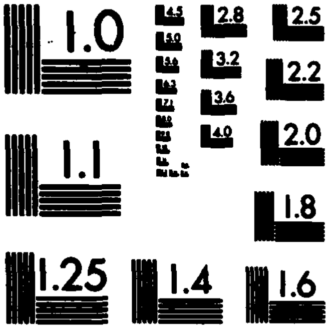
An I-frame destined for a location beyond the current receive window is an exception and the I-frame is discarded.

AD-A120 771   COMMUNICATIONS PROTOCOL STRUCTURE(U) CONTEL INFORMATION   2/2
SYSTEMS VIENNA VA 23 SEP 82 FR.343.01 SBI-AD-E001 358
N00014-81-C-2477

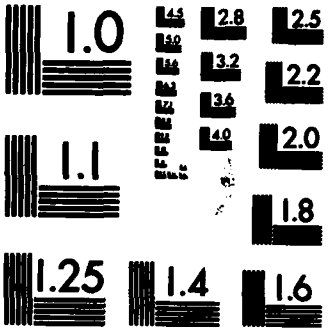
UNCLASSIFIED   F/G 17/2   NL

END

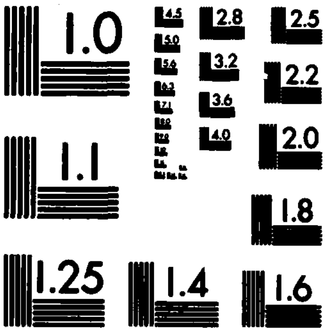
FILMED



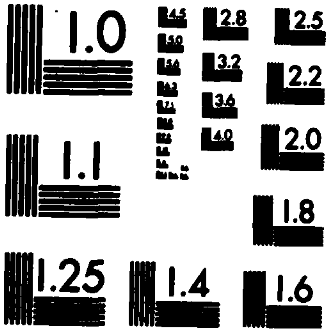
MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A



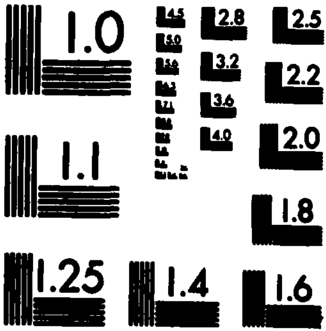
MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

7.6.3 Advance Receive Window

The receive window is advanced forward if the first I-frame (Left Receive Window) in the receive window has been received and ACKed. The first I-frame leaves the receive window and is forwarded to the network layer. The receive window continues to advance forward until the first location of the receive window is empty.

7.6.4 Update Send Window

The ACKs, are used to inform the send window which I-frames were received by the receiver. The same processing applies to both the S-frame acknowledgment as well as the piggybacked acknowledgment in an I-frame. If the ACK bit map indicates a frame has been received, the sender discards the frame and updates the status of the frame to indicate so. Also the ACK bit map may indicate that a frame has not been received even before the corresponding timeout for that frame expires. In this case the frame is transferred from the retransmission queue to the transmission queue. For example, if the ACK bit map from the receiver contains ACKs for I-frames 3, 4, 6 and 7, the sender may infer I-frame 5 has to be retransmitted. However, acknowledgment information delivered over the communications channel may not be as current as the send state; hence a frame would not be so transferred from the retransmission queue to the transmission queue unless it had been held in the retransmission queue a minimum amount. In summary the sender need not necessarily have to wait for a timeout before retransmitting an I-frame.

Thus the timeout for frame 5 would be purged, and frame 5 would be transferred from the retransmission queue to the transmission queue. The link level will advance the send window if the first I-frame in the send window has been received by the receiver. The I-frame leaving the send window, causes the network layer to pass a block, if any, to the link level. The send window will continue to advance until either the first I-frame in the send window has yet to be received by the receiver or has yet to be sent to the receiver.

7.6.5 Radio Silent Reception At A Ship

A ship in this mode, receives I-frames and S-frames only. It does not receive the Data Exchange Poll. Such a ship cannot transmit frames to the NCS. This means that the ship cannot ACK any I-frames sent by the NCS.

Since a ship which goes from the normal mode of communication to the radio silent mode, may have a non empty receive window, the I-frames in the receive window are simply forwarded to the network level without advancing the receive window.

Any I-frames or S-frames received, by the ship's link level, are simply forwarded to the network level. The concept of a receive window is no longer necessary.

It should be noted that at the end of radio silent reception, the send window at the NCS and the receive window at the ship will be out of synchronization unless steps are taken to prevent this from occurring. Thus, when a ship in radio silent mode wants to re-enter the net, it must execute the normal net entry algorithm.

7.7 Retransmission of I-frames

The sender records the time an I-frame is transmitted to the receiver. If an ACK for the I-frame is not received, from the receiver, within a predefined interval of time, the sender retransmits the I-frame to the receiver. Also, the sender will retransmit an I-frame if it infers that the receiver did not receive the I-frame (see Section 7.6.4).

When the timeout expires for an I-frame, the link level process checks how many times the I-frame has been transmitted. If the I-frame has already been transmitted the maximum number of times, the code rate will be decreased and the retransmission process repeated. If the code rate cannot be decreased, an exception condition is identified to the higher level protocol.

Otherwise the link level process will transfer the frame from the retransmission queue into the link transmission in the appropriate order for subsequent transmission.

7.8 Net Exit

Network exit is a part of the high level protocol. A ship desiring to exit a network ensures that all its application processes communicating with application processes at the NCS have no outstanding messages. The ship then informs its link level to disconnect the link. The ship's link level will send a disconnect command to the NCS link level. The format of the disconnect frame is shown in Figure 7(c).

The link level, at the NCS, on receiving a disconnect command from a ship will send an acknowledgement to the ship's link level. The link level at the ship, on receipt of this acknowledgement, will release send/receive sequence lists, all queues, ACK bit map and remain in the disconnected state. The link level at the ship on failing to receive an acknowledgement will retransmit the disconnect command.

The link level at the NCS, on receiving a disconnect command from a ship, will release send/receive sequence lists, all queues and ACK bit map assigned to the ship. The link level will inform the high level protocol at the NCS that the ship has left the network. The access level at the NCS will remove the ship from the Poll List.

7.9 Application Process Interface

Application processes requesting transmission of messages to the receiver, queue these messages at the Output Queue. If the Output Queue is full, the application processes will not be able to queue their messages.

8. ADAPTIVE CODE RATE

The recommended strategy for dynamically selecting code rate in the initial implementation is to:

- Allow the ship radio operator to request a code rate for both transmission and reception at net entry; this would be incorporated in the Net Entry Request frame.

- Allow the NCS to override this request in the Net Entry Initialization frame; this would be done automatically without NCS operator intervention; however it could be based on ship co-ordinates (relative to the NCS).

- Provide the mechanism as described in Section 7 to decrease the code rate when an expected acknowledgment is not received after a specified number of iterations.

- All net entry frames are transmitted at an information rate of 300 bps.

- All frames directed to radio silent terminals are transmitted at an information rate of 300 bps.

The recommended enhancements for immediate implementation is to provide the capability:

- For an operator to manually trigger the access level to send a frame to the station with which it is communicating requesting that the station increase or decrease its code rate.

The next level of recommended enhancements is to provide the capabilities:

- For each receiver to maintain a moving average of the decoding metric for its station with which it is communicating.

- Whenever the moving average crosses either an upper or lower threshold, automatically trigger the access level to send a frame requesting the opposite station to correspondingly adjust its code rate.

In general this problem should be further studies in terms of quantitative performance. The substance of the problem is to establish upon net entry the rate at which each radio should transmit. Because of the relative slow rate of changes in the environment (relatively slow movement and no jamming), it is anticipated that once an optimal code rate is established, it will not have to frequently be adjusted. Hence for the immediate operation the manual selection of code rates should be satisfactory.

9. FLOW CONTROL

Buffer congestion at the NCS is not really a problem because of the low bit rate and because the NCS controls the rate at which it receives transmissions from the ships. However, the NCS does impose some flow control restrictions on the ships. It does so by limiting the number of frames a ship is allowed to transmit to the NCS when the ship is polled.

Flow control at the ships occurs through the use of the RR/RNR field in the I-frames transmitted by the ship to the NCS (see Figure 7(a)). When the occupancy of the ship's buffers reach a threshold value, the ship will present a RNR (Receive not ready) in the RR/RNR field (value contained in this field is 1). This is an indication to the NCS that the ship's buffers are full and the NCS suspends transmission to the ship.

When the occupancy of the ship's buffers fall below a threshold value, the ship will present a RR (receive ready) in the RR/RNR field (value contained in this field is 0). This is an indication to the NCS that the ship's buffers can receive more frames and the NCS resumes transmission to the ship.

The mechanism to trigger this flow control method is dependent upon the buffer management algorithms employed. Since this is properly an implementation issue, detailed definition of a flow control algorithm is beyond the scope of this report.

APPENDIX A

PROTOCOL SPECIFIC PARAMETERS

The protocol, described in this report, has been found to contain several parameters. A list of these parameters is included here. Their definitions could be found in the text of this volume.

The following is a list of the parameters and their recommended values in parentheses:

1. Maximum number of ships in the network (256).
2. Duration between Net Entry Phase initiation requests (2 minutes).
3. Minimum number of time slots (2).
4. Maximum number of time slots (8).
5. Duration of a time slot (2 ms).
6. Size of Sequence Space (32).
7. Size of Window (16).
8. Duration after which an I-frame is retransmitted (30 sec).
9. Maximum number of retransmissions (5).
10. Weighting factor of relative importance of priority 1 blocks to priority 2 blocks (.7).
11. Duration of the poll cycle (2 minutes).
12. Number of times a Data Exchange Poll may be retransmitted (3).
13. Number of times a ship attempts Net Entry (5).
14. Duration of time, after which, a lack of response from the receiver will initiate recovery action (N_B/(2 x code rate) seconds).
15. Number of times a sender sends a S-frame to invoke a response from the receiver, before presuming the receiver is dead (5).
16. Minimum time in retransmission queue (2 seconds).

CONTEL
INFORMATION SYSTEMS

VOLUME III

VOICE/DATA INTEGRATION STRATEGIES

1. INTRODUCTION

In this volume, we address the problem of integrating voice and data traffic into future digital Naval Telecommunications System (NTS) networks. The NTS networks will operate in an environment of rapidly changing network topology caused by:

- Node mobility due to movement of ships.
- Node destruction by hostile forces.
- Degradation of link performance due to jamming.

The network may carry traffic such as record traffic and commands, weapons control and guidance traffic, surveillance, position location, voice, and graphics/digital FAX. The traffic has different service requirements:

- Priorities, including routine, priority, immediate, flash, and flash override.
- Secure transmission of information.

Furthermore, the source and destination sites may be located thousands of miles apart.

The transmission channels that may be potentially employed include both HF terrestrial channels or EHF satellite channels. However, in both cases it is expected that only limited channel capacity, e.g., 2400 bps, may be made available to individual ones. Since the future NTS will be based on digital transmission techniques, the NTS network could carry both voice and data traffic. It is especially attractive to efficiently integrate voice and data traffic on the same network, because of the limited available transmission bandwidth. Since voice traffic is characterized by silent and talk periods, the silent periods may be multiplexed with data for efficient bandwidth utilization. This working memorandum discusses the strategies available for integrating voice and data traffic over the NTS network.

2. STRUCTURE OF THE PROBLEM

The integration of diverse traffic types such as voice and data will have a major impact on the entire NTS network. In fact, such integration will permeate all levels of protocols. However, the key aspects of a voice/data integration strategy are the resource sharing and switching techniques. In Section 3, we categorize the range of alternatives considered. Then in Section 4, we describe the strategies deemed most appropriate for NTS and evaluate their relative merits.

Also, another area where voice/data integration will have a major impact is on the Presentation layer protocol (Layer 6 of the ISO Open Systems Interconnection Reference Model). This layer includes the voice digitization technique, whose rate may be dynamically varied, and the voice packetization and reassembly algorithms. The issues associated with these techniques are discussed in Section 5.

In conclusion, we enumerate the issues for further study in Section 6.

3. VOICE/DATA INTEGRATION STRATEGY CHARACTERIZATION

A categorization of the switching techniques is illustrated in Figure 1. Voice/data could be circuit switched as in the traditional methods, or it could be packet switched, alternatively voice/data could also be hybrid switched (a combination of packet switching and circuit switching).

In circuit switching, a dedicated path through the communications network is allocated to each pair of communicating subscribers. The circuit switching methods include: traditional circuit switching, voice circuit switching such as Common Channel Interoffice Switching (CCIS), fast circuit switching and Time Assigned Speech Interpolation (TASI) or Adaptive Data Multiplexing (ADM).

In traditional circuit switching, a complete end-to-end circuit is established for each pair of voice and data users and dedicated for the full duration of use. A subscriber places a call to the destination subscriber. The switches then sets up a dedicated physical connection between the subscribers, consisting of a sequence of point-to-point circuits, joined together by switches at the junctions.

Traditional circuit switching uses the inband channel for the signaling necessary to control voice calls (call set up, disconnection) on a per circuit or trunk basis. With CCIS, the signals for the control of voice trunk groups have the format of digital data packets traveling in a store-and-forward mode via a separate network using an out-of-band channel. This results in a substantial drop in the set-up overhead and also allows the implementation of priority-preemption, conference calls, etc.

Performance improvement can be further obtained by using traditional circuit switching with TASI for voice traffic and ADM for interactive data traffic. With TASI, voice channels are allocated to calls only when these are active. Thus, a fixed number of voice channels can be multiplexed among a greater number of calls. ADM is similar to TASI where a common channel signaling protocol squeezes more channels by exploiting the idle intervals in a data communications session.

Fast circuit switching is used for interactive data users. This method causes a circuit to be established for every message which is to be sent, and then disconnected after transmission. It takes advantage of the low duty cycle of interactive users by not dedicating the circuit to the user during his "think-time." The recovered capacity could be used to transfer more data, thus improving network efficiency.

In packet switching no dedicated path is allocated between pairs of communicating subscribers. Instead, the available transmission capacity is simultaneously shared between

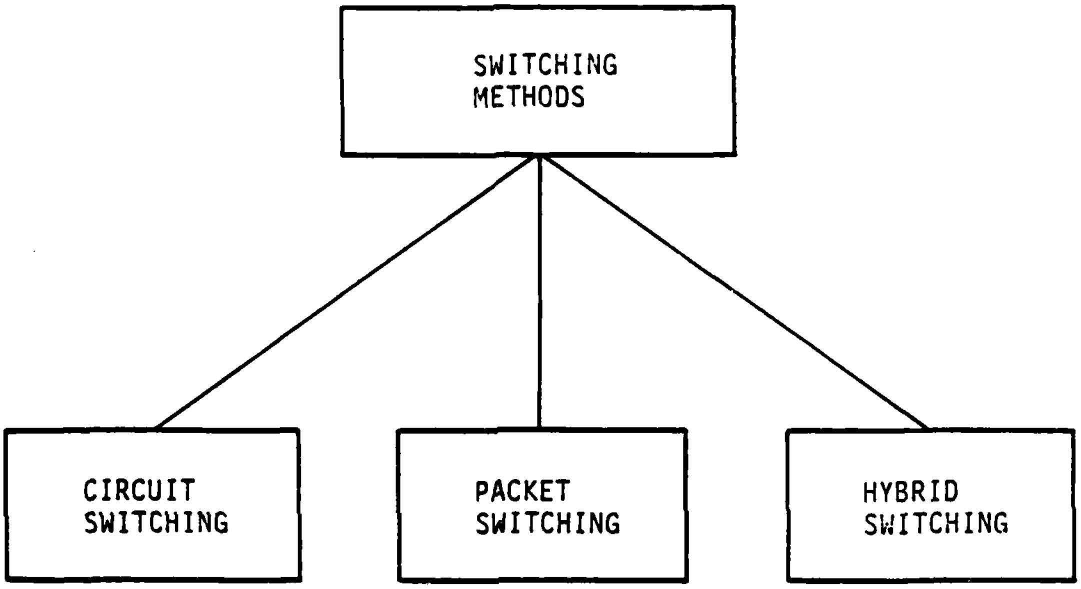


FIGURE 1: SWITCHING TECHNIQUES

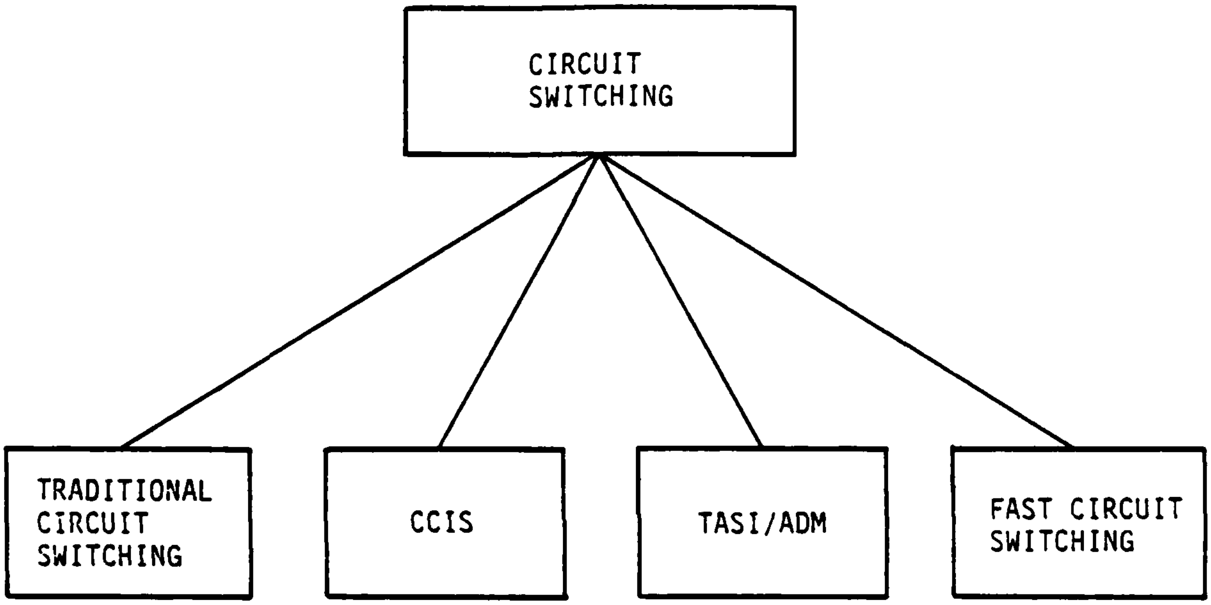


FIGURE 1: SWITCHING TECHNIQUES (CONTINUED)

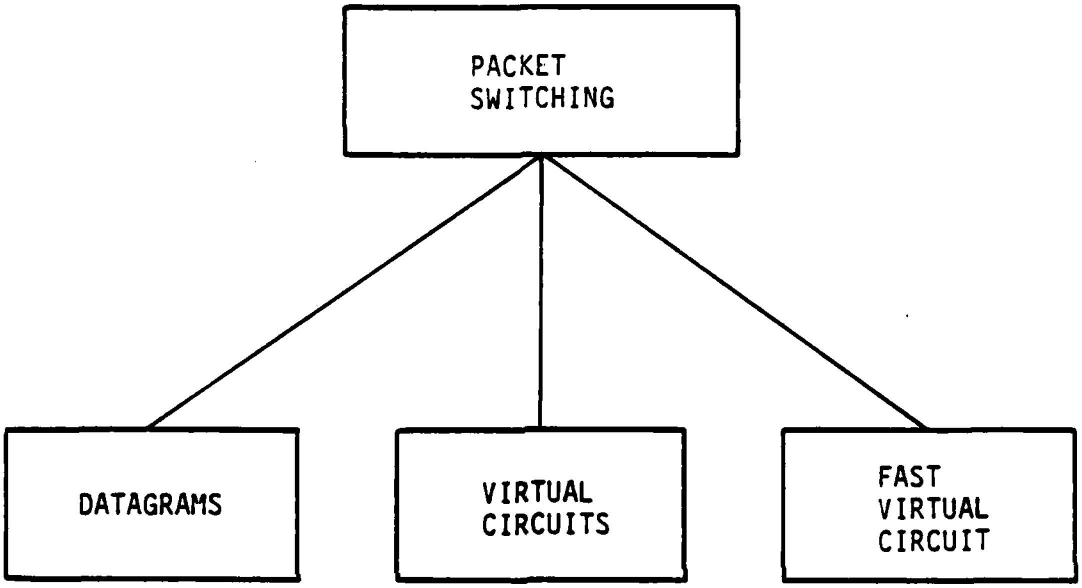


FIGURE 1: SWITCHING TECHNIQUES (CONTINUED)

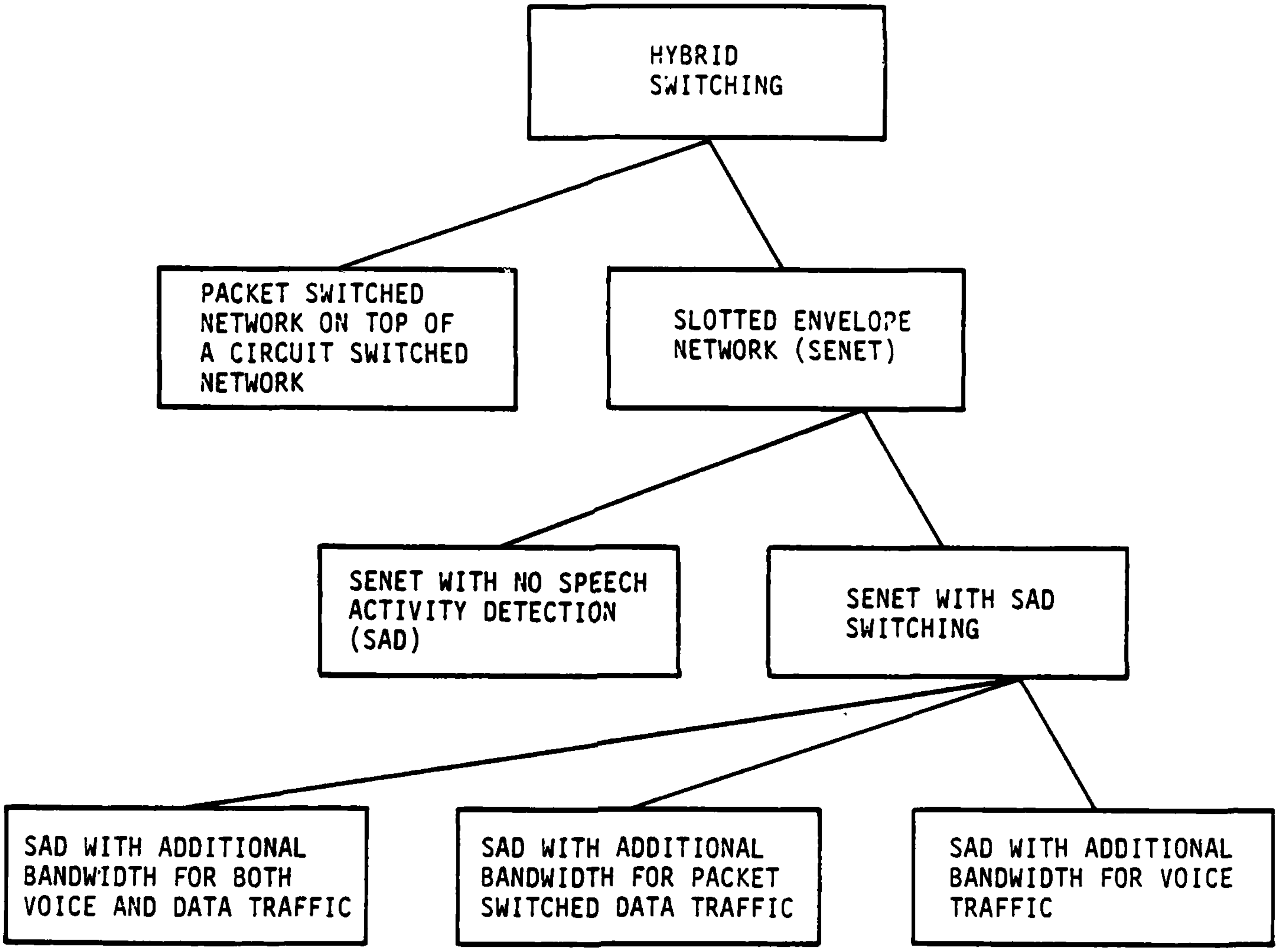

HYBRID SWITCHING
PACKET SWITCHED NETWORK ON TOP OF A CIRCUIT SWITCHED NETWORK
SLOTTED ENVELOPE NETWORK (SENET)
SENET WITH NO SPEECH ACTIVITY DETECTION (SAD)
SENET WITH SAD SWITCHING
SAD WITH ADDITIONAL BANDWIDTH FOR BOTH VOICE AND DATA TRAFFIC
SAD WITH ADDITIONAL BANDWIDTH FOR PACKET SWITCHED DATA TRAFFIC
SAD WITH ADDITIONAL BANDWIDTH FOR VOICE TRAFFIC

FIGURE 1: SWITCHING TECHNIQUES (CONTINUED)

pairs of communicating subscribers. Three packet switching techniques described here include: datagrams, virtual circuits and fast virtual circuits. In the datagram mode, the voice and data traffic is formed into separate packets. Packets (voice or data) are routed adaptively at each node, which selects the output trunk for each newly arrived packet. This selection is based on the destination and the current values of estimated delays stored in the routing tables.

In the virtual circuit mode, a path through the network is determined for each connection when initiated, and resources, e.g., buffers, may be allocated at this time. Tables are maintained at all intermediate nodes, as well as at the source and destination nodes, in order to route all subsequent packets on the same circuit. Thus, all packets on a connection follow the same path, and sequential delivery is assured.

The flow of information is restricted to occur after a virtual circuit has been set up. This is a considerable overhead if the information consists of only few packets. To speed up the process, the fast virtual circuit mode is used. In this mode, the call set up packet also carries the data information to be transferred. In the virtual circuit mode the call set up packet carries no data portion.

Hybrid switching is a combination of circuit switching and packet switching. There are several ways of integrating circuit and packet switching. One method is to use circuit switching for setting up paths in the network and then use packet switching for moving data along the paths. The output of the packet switch is treated by the circuit switch as a regular subscriber and an end-to-end channel (physical, FDM band or TDM slot) is dedicated to the subscriber.

Another method for integrating circuit and packet switching is to divide the trunk capacity between the two. The slotted envelope network (SENET) is an approach to do this. In brief, SENET is a framed time-division multiplex scheme in which frames are divided into slots which may be allocated to voice on a circuit-switched basis or to data on a packet-switched basis.

The trunk capacity allocated to the circuit switched and packet switched portion could be fixed (necessarily so in the absence of speech activity detection or SAD). This technique results in unassigned voice slots during the silence intervals. If SAD is present, then the voice slots could be assigned only during the talkspurts and when silence is detected the voice slots could be assigned for transmitting data packets. This results in the dynamic allocation of trunk capacity between the circuit switched and packet switched portion.

The dynamic allocation of trunk capacity can be done in several ways. One method is to allow either the circuit switched portion or the packet switched portion to use the other's unused capacity when needed. A second method allows only the packet switched portion to use the unused capacity in the circuit switched portion but not vice verse versa. Finally, a third method allows only the circuit switched portion to use the unused capacity in the packet switched portion but not vice versa.

Traditionally, circuit switching has been used for voice and packet switching for data. However, alternative schemes including such novel approaches as packet switching for voice and interactive data and circuit switching for bulk data (file transfer, graphics, and facsimile) may be more efficient. A voice/data integration strategy would include the switching technique and associated network for each applications. Clearly, various combinations of traditional and novel schemes are appropriate for consideration in NTS.

In this volume four of the techniques, described above, are isolated for further analysis. These include:

Circuit switch using TASI.
Packet switching using virtual circuits.
Packet switching using datagrams.
Hybrid switching using SENET with movable boundary.

4. SWITCHING ALTERNATIVES

In this section we present and evaluate four mechanisms for integrated voice/data communications. The four mechanisms are: circuit switching with TASI, packet switching using either virtual circuits or datagrams, and hybrid switching with SENET.

Voice communication has traditionally been effected through the use of dedicated circuits with fixed transmission capacity. Measurements on normal conversations have shown that an average speaker is active approximately 40 percent of the time [1]. This means that 40 percent of the transmission capacity is used by the talkspurts in calls while the remaining 60 percent is unused due to the silence periods in the calls. In more advanced circuit switching, this idle capacity is put to good use via a mechanism called TASI (which will be described in Section 4.1.1). In packet switching (described in sections 4.1.2 and 4.1.3) voice packets are generated only during the talkspurts in a call so that the transmission capacity is available for other voice/data packets during the silence periods. In hybrid switching with SENET (described in section 4.1.4), voice channels are used during the talkspurts in a call while during the silent periods the channels are released for use by either other voice subscribers or data packets.

A description of TASI, packet switching, and SENET is given in the following sections. The criterion used for the evaluation of the techniques is given in section 4.2. The three concepts are evaluated in section 4.3. Finally, results of the evaluation are presented in section 4.4.

4.1 Descriptions

4.1.1 Circuit Switching Using TASI

In traditional circuit switching, a complete end-to-end circuit is established for each pair of voice users and dedicated for the full duration of use. The caller sends a call set up message, on the CCIS channel, to the callee. This signal contains the caller's address, the callee's address, and control information. The message is routed through the circuit switched network until it reaches its destination (the callee). The nodes, which this message traverses, set up the circuit by connecting the incoming channel, to the node, with the outgoing channel, from the node. The destination node (the callee) informs the source node (the caller) that it is ready to listen, by sending, to the source node, a call confirmation message on the CCIS channel. This message contains the caller's address, the callee's address, and control information. Receipt of this message by the caller is an indication that the circuit is set up and voice transmission may begin. The caller or callee will send a call breakdown message, on the CCIS channel, whenever the conversation is terminated. This message will follow the established circuit and inform the intermediate nodes it traverses that the circuit may be broken. This message contains such items as the caller's address, the callee's address, control and accounting information.

Traditional circuit switching of voice has a poor transmission efficiency because of the idle periods during the conversation. This may be improved by using the traditional circuit switches and supplementing them with TASI. This concept is illustrated in Figure 2. TASI is a well known technique in which the idle time during conversational voice calls is used to accommodate additional calls. With a sufficiently large number of channels, most of the idle time on the channels can be filled, giving an enhancement in transmission capacity greater than two-fold.

Observations on voice channels have indicated that speech is actively present in a busy channel about 40% of the time. Each period of time occupied by a caller's speech is called a talkspurt. The low activity is largely due to the fact that a speaker talks less than half the time during a conversation and that each speaker's speech is carried on a separate channel.

TASI exploits the low talkspurt activity by assigning channels only when a talkspurt is present. The process becomes more efficient as the number of channels increases. An attempt to interpolate two independent conversations on a single channel will cause a large percentage of the speech to be lost due to competition for simultaneous channel occupancy. When a large number of independent conversations compete for some smaller number of channels, the same competition prevails and there is always a finite probability that the number of conversations demanding service will exceed the number of available channels. This competition manifests itself in the form of clipping of the initial portion of a talkspurt, or competitive clipping. The portion of time that speech is lost due to such competition is called the cut-out time fraction. Provided that the population of incoming calls is sufficiently large and the ratio of the number of incoming calls to the number of channels available for transmission (the TASI advantage) is sufficiently small, the fraction of speech that is cut out can be rendered acceptably small.

In addition to this competitive clipping, additional care must be taken to minimize connect clipping. At each TASI processor, the presence of speech on an incoming channel is sensed by a speech detector which indicates a request for an outgoing channel. A channel assignment processor (sender) assigns an idle outgoing channel to the incoming channel and sends a connect message, on the CCIS channel, to the TASI processor at the other end of the

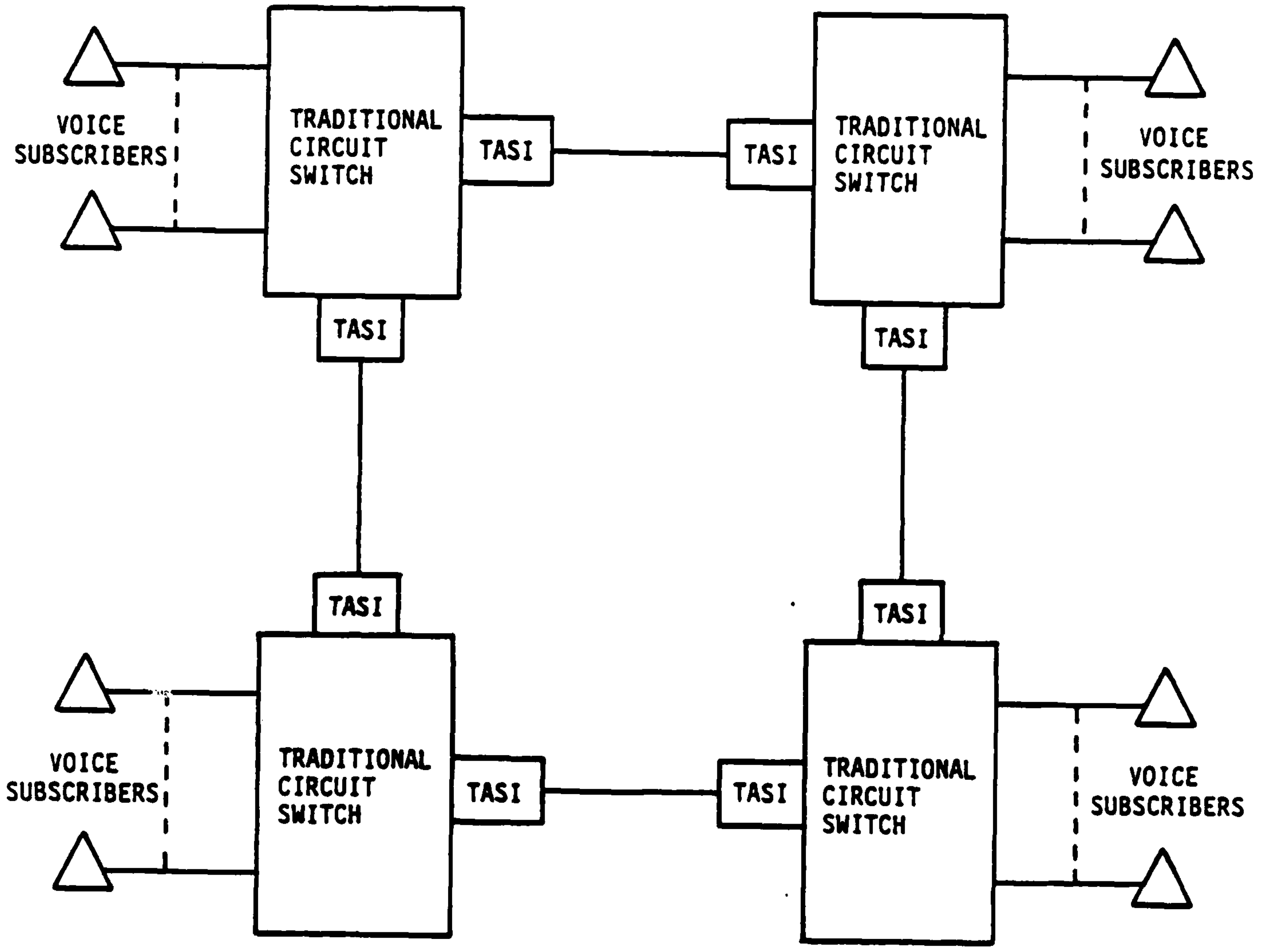


FIGURE 2: TASI WITH THE TRADITIONAL CIRCUIT SWITCH

assigned outgoing channel. The connect message contains such information as the circuit identifier and the outgoing channel number. During the time required to assign channels ana reestablish the circuit (thus connecting the caller with the callee), the speech can be clipped. This phenomenon is called a connect clip and is to be distinguished from the competitive clip discussed previously. The connect clip, at each node, is on the order of 10 milliseconds (p. 355 of Reference 2).

The connect clip is constrained to a very short duration to minimize subjective degradation consistent with reliable connect operation. The statistics of this tvpe of clip are controlled by the competition of connect messages (e.g., other CCIS messages) for space on the CCIS channel. During periods of light loading, channels may remain assigned, to a circuit, even during idle intervals so as to minimize the effects of connect clipping. However, a demand for use of an idle outgoing channel, will be met, by a node, and a disconnect message sent to the node at the other end of the idle outoing channel, indicating that the idle outgoing channel will be reassigned to another circuit. The disconnect message is sent in the CCIS channel and contains such information as the circuit number and the idle outgoing channel number. An outgoing channel is reassigned to the disconnected circuit when the node at the sending end of the channel detects a talkspurt on the incoming circuit channel. This is done by the connect message.

All of the messages, mentioned above, use the CCIS channel. These messages are error controlled and use the CCIS format. These messages are handled as packet switched data.

This procedure may be repeated at each node in the circuit (defined by the call set up message) or only at nodes in the circuit where the circuit has been disconnected. However, this technology is yet to be demonstrated.

4.1.2 Packet Switching Using Virtual Circuits

When a voice connection is requested, a path is established through the network. A call request packet is addressed to the destination with the requested bandwidth and a connection ID. The call request packet will be routed though the network to the destination. The routing decision may be made by considering the available bandwidth on the outgoing trunks and by comparing this to the bandwidth requested for the connection. An algorithm will determine whether a connection can be established through the outgoing trunks. The criterion will be to minimize the probability of discard of the traffic when the packets (comprising the traffic) encounter congested nodes on their virtual circuits. This congestion may be due to, for example, the simultaneous use of a particular segment by several different connections. The criterion is related to the concept of grade of service whereby the grade of service is improved by reducing the probability of discarding traffic and vice versa.

As the call request packet passes through each node, a data structure is created containing all specifics of the connection and indicating the trunk on which the call request packet left the node. If the call request packet is blocked in its route, the blocking node may send a blocking notice to the source, or the connection may time out. If the call request packet reaches the destination node, the destination node prepares a call confirmation packet and dispatches it to the source node. This packet may return along the path already established or it may return along a different path. This discussion of virtual circuit set up illustrates a sample approach; and other methods of setup may employ slightly different procedures.

When the call confirmation packet reaches the source node, the connection is considered established. Then, a waiting voice subscriber is notified of the connection. The voice packets will contain a connection ID in the header. The connection ID may be the sequence number of the connections on a link or it may be unique in the system. In any case, the header will be much shorter than that ordinarily required in a packet network. When the packet is to be routed, the connection ID allows the node to access the stored connection table for queueing, security and the output trunk for routing. All subsequent packets in a connection have an abbreviated follow-on header, thus improving transmission efficiency, and are routed sequentially, thus guaranteeing properly sequenced delivery. Voice connections are explicitly terminated at the source node when the caller hangs up. A call disconnect packet is sent to the destination node. This packet follows the route of the call request packet and informs all the nodes it traverses that the call is disconnected.

Once a connection is set up, no error correction of the voice packets takes place. So the voice packets need not contain bits to perform error correction. However, the voice packets do need bits to detect errors in headers. All signaling packets are fully error controlled and are handled as all other packet switched data.

In summary, a voice packet contains the following overhead:

- Flags.

- Connection ID.

- Error detection bits.

Even though packets will generally not arrive out of order, the inter-arrival time between packets is variable. Hence, an algorithm for reassembly of the packets into an intelligible conversation is required. Some of these algorithms may require additional header information such as time stamping; these issues are discussed further in Section 5.

4.1.3 Packet Switching Using Datagrams

The previous section (Section 4.1.2) described packet switching with virtual circuits. It is also possible to send voice as well as data packets in a packet switched system using datagrams. This mode is more suitable in a hostile environment where the network topology is very dynamic and jamming is prevalent.

In this approach, the voice packets (from a talkspurt) are independently routed to the destination. No path is set up for the duration of the call. Each packet is transported across the network to its destination independently of other packets for the same connection. Individual packets can be alternately routed as appropriate. This could result in packets belonging to the same talkspurt being received out of order because of the varying delays encountered over different network paths. Hence, a reassembly mechanism is required at the destination node which is more sophisticated than the algorithm required for virtual circuits. Likewise, the algorithm for adaptive routing of packets is more complex. However, communication is more robust in the event of failure since voice packets can be routed around failed paths.

The packet header is large because it includes the full source and destination addresses. Because of this, the information text is large to offset the overhead inefficiency introduced by the large header. For long packets, network delivery delay could be excessive. Also lost or excessively delayed voice packets now become a system problem.

No error correction of the voice packets takes place. So the voice packets need not contain bits to perform error correction. However, the voice packets do need bits to detect errors.

In summary, a voice packet contains the following overhead:

- Flags.

- Source address.

- Destination address.

- Sequence number.

- Error detection bits.

4.1.4 Hybrid Switching Using SENET with Movable Boundary [10]

In the hybrid switching concept, switching and transmission facilities of the network are shared between traffic, with a single facility providing both circuit and packet switched modes of operation. The transmission capacity of the link between two nodes is used to carry circuit switched traffic at one instant and packet switched traffic at another instant. The SENET multiplexing scheme, which uses a TDM master frame, partitions link capacity into two regions: a circuit switched synchronous region and a packet switched asynchronous region.

The circuit switched operation could be traditional or using TASI as described in Section 4.1.1. A complete end-to-end circuit is established for every voice user and dedicated for the full duration of use. Signaling messages are used to convey the information needed for the network to interconnect one subscriber with another. These messages are error controlled and handled as packet switched data.

During call set up, the circuit switched slot (in an incoming master frame) is assigned, by a node, to a circuit switched slot (in the master frame of the apropriate outgoing link). This assignment is forwarded by signaling messages to the next node. The frame structure for the outgoing link is also communicated to the next node. Once an end-to-end connection is made through the network, the maps at each node specify the incoming link, frame and slot of each call with the appropriate outgoing link, frame and slot. Since error detection/correction is not applied to circuit switched calls, no additional processing by the nodes is necessary once the connection is established.

In packet switched operation, messages are divided into packets for transmission through the network. Each packet has additional bits added for address and administrative purposes (e.g., the origination/destination addresses, sequence number, priority, etc.). The outgoing link for an incoming packet is determined by the virtual circuit or for datagram by processing the header, and is a function of the destination node, priority, routing strategy, etc. Received packets are queued at the outgoing link for transmission on a subsequent frame. Packets are error checked along the way, e.g., each time another wideband link is traversed. At the destination, the node reassembles the packets into complete messages, which are then presented to the host.

Transmission capacity is assigned to the different regions by the moving boundary method, which allows both types of traffic to coexist on the same link according to a flexible boundary rule in which transmission capacity is dynamically shared between the two regions. While a boundary is normally assigned between packet and circuit switched regions, one type of traffic can utilize idle channel capacity normally assigned to the other.

In the implementation of this scheme the SENET frame starts with a start-of-frame (SOF) marker followed by the circuit switched region. The circuit switched region is followed by the packet switched region. The remainder of the frame is unused capacity. Allocation maps are maintained at both ends of each link. These maps contain call identity, starting point of the connection in the frame, magnitude of the capacity used by the connection and precedence of the connection. Such maps are maintained for each outgoing and incoming trunk in the node.

Virtual channel allocations in the circuit switched region are made following the last allocation. As calls terminate, all subsequent allocations are moved forward, in the frame, toward the SOF marker by the magnitude of the terminated allocation. A single signaling message will be forwarded along the path of the terminated call to update all of the allocation maps. Other signaling messages will be propagated along paths to update the allocation maps of each affected node.

The remainder of the frame capacity is used for packet switched data. There is no boundary marker between the circuit and packet switched regions. The signaling messages used to effect this scheme could use the packet switched region or a dedicated slot serving as an orderwire.

The circuit switched region could use TASI and take advantage of the inherent properties of speech. During the silence periods of a conversation, transmission ceases, and the available capacity can be used to transmit either other conversations or data packets. A control vector is located next to the SOH marker, in the circuit switched region of the frame. This vector indicates which user slots are in talkspurt and which are in silence. Virtual circuits in which the voice source is in the silence period can be used to carry other conversations or data packets. Voice sources that leave the silent state and resume speech generation may be "frozen out," i.e., they may not find an available slot in which to transmit. A sufficient number of slots must be provided to ensure that the percentage of speech "frozen out" is within an acceptable percentage.

Data packets usually have a short service duration while voice traffic usually has very long holding times. Consider a case where the data arrival rate is sufficiently large that it requires some voice capacity to remain stable. Occasionally all voice channels will be occupied with voice traffic and the long voice holding time indicates that this condition will be maintained for a substantial period of time (i.e., the maximum allowable voice channels in a frame will be used over several consecutive frames). Under these conditions, Gaver and Lehoczky [3] have pointed out that since the data arrival rate is high enough to require voice channels, for meeting stability requirements, the lack of any available voice channels leads to a buildup of data packets. And since the available voice channel shortage extends over several consecutive frames, these data packet queues will tend to be very long and will take a substantial time to work off. As a result, the node must have adequate buffer capacity to hold these very long data queues or else data packets will be lost. In addition, these long data queues cause the average delay for the packets to become large. Another alternative to the solution of this problem is to reduce the voice capacity and consequently increase capacity for the data packet traffic.

Real time traffic (such as interactive traffic) cannot tolerate such long delays. As a result, real time traffic cannot be used in situations described in the previous paragraph. However, non-real time traffic (such as file transfers, facsimile, etc.) can easily withstand the problems mentioned in the previous paragraph. This is true because this kind of traffic requires considerable time to be transmitted and thus is not greatly affected by the delays.

4.1.5 Summary of Fundamental Differences

The fundamental differences between the various switching schemes described so far can be compared by their use of control messages. The control messages in circuit switching with TASI use the out-of-band channel. These messages occur not only during call set up and disconnection but also during voice transmission (namely before certain talkspurts). The control messages in packet switching (datagram or virtual circuit) use the in-band channel; these messages may be separate messages as in call setup or included in the header. There is no out-band channel. In hybrid switching (using SENET with movable boundary), the control messages, generated by traffic on the circuit switched portion, use the packet switched portion of the frame. A separate slot could be dedicated as an orderwire (out-of-band) or control messages could be sent via a header or using a slot also employed for data.

4.2 Evaluation Criterion

The criterion used to evaluate the three concepts for voice communications are defined below.

4.2.1 Overhead Efficiency

Overhead is defined as the transmission bandwidth required by the voice traffic in excess of that required for transmitting pure voice.

4.2.2 Fractional Speech Loss

This is a measure of the speech lost due to freeze-outs in the circuit switching with TASI case or due to excess queueing delays in the packet switching case (using either virtual circuits or datagrams).

4.2.3 Precedence/Preemption

Precedence is defined as the ability of the higher priority traffic to preempt low priority traffic and thus be serviced before the low priority traffic. Traffic, that is preempted may be abnormally terminated (as in the case of circuit switched voice calls) or suspended (as in the case of packet switched voice calls). Suspended calls can be resumed once a capacity becomes available. Abnormally terminated calls need to be reinitiated.

4.2.4 Voice Continuity

This is a measure of the smoothness of the received voice. Received voice having silent gaps (arising due to the technique used) which are noticeable is said to have poor voice continuity.

4.2.5 Processing Load/Resource Requirements

This criterion is used to compare the resources required for implementing the voice communciation techniques. In addition, the processing load at the nodes is compared.

4.2.6 Error Control

This criterion is used to guage the robustness of the protocol under error conditions. Specifically, a comparison is made of the extent of performance degradation under error conditions.

4.3 Analysis

The four concepts for integrating voice and data traffic are analyzed using the evaluation criterion defined in the previous section.

4.3.1 Overhead Efficiency

4.3.1.1 Circuit Switching with TASI

Let

T_i = length of call i in seconds

p = fraction of time speech is present in a call

then

duration of talkspurts in a call = pT_i seconds.

Let

T_t = measured value of effective talkspurt in seconds

then

number of talkspurts in call i = $\frac{pT_i}{T_t}$

and

number of connect messages in call i = $\dfrac{apT_i}{T_t}$

where

a = proportion of talkspurts requiring connect messages.

The parameter, a, is a function of the traffic offered to the network. If the traffic volume is low, then a channel may be dedicated to a conversation for a long period of time, and thus there will be few correct messages. Alternatively, if the traffic volume is very heavy, a connect (and disconnect) message will be required for nearly every talkspurt. Assuming that talkspurts can be buffered (so that no competitive clipping occurs), M. Fischer [4] has obtained equations for the expected number of talkspurts in a queue $E[Q_t^q]$ and the expected number of talkspurts in the system $E[Q_t]$.

Then the parameter, a, can be estimated

$$a = \frac{E[Q_t^q]}{E[Q_t]}.$$

However, this is an approximation to the TASI system.

Also, number of disconnect messages in call i = $\dfrac{apTi}{T_t}$.

Therefore,

number of connect + disconnect messages in call i = $\dfrac{2\ apTi}{T_t}$

and

number of connect + disconnect messages for n calls = $\dfrac{2\,a\,p}{T_t} \displaystyle\sum_{i=1}^{n} T_i$.

But

$$T_A \quad = \quad \text{average length of a call} \quad = \quad \frac{\sum_{i=1}^{n} T_i}{n}$$

then

$$\sum_{i=1}^{n} T_i = nT_A$$

and

$$\text{number of connect} + \text{disconnect messages for n calls} = \frac{2\ apnT_A}{T_t}$$

or

$$\text{number of connect} + \text{disconnect messages per second for n calls} = \frac{2\ apn}{T_t} .$$

If

$$L \quad = \quad \text{average length of a connect/disconnect message including header}$$

Then the CCIS channel capacity required to transport these connect/disconnect messages is

$$O_{TASI} = \frac{2\ a\ pnL}{T_t} \text{ bits per second.}$$

4.3.1.2 Packet Switching Using Virtual Circuits or Datagrams

Let

R_t = speech digitizer rate in bits per second

T_t = measured value of effective talkspurt in seconds

then

number of bits in a talkspurt = $T_t R_t$.

If

L_A = average length of a voice packet in bits

then

number of packets in the talkspurt = $\frac{T_t R_t}{L_A}$.

It was previously shown that the

number of talkspurts in call i = $\frac{pTi}{T_t}$

then the

$$\text{total number of talkspurts for n calls} = \frac{p}{T_t} \quad \frac{\sum_{i=1}^{n} T_i}{n}$$

$$= \frac{pnT_A}{T_t}$$

or

total number of talkspurts per second for n calls $= \frac{pn}{T_t}$.

Thus, it follows that

the number of packets per second for n calls $= \frac{pn}{T_t} \; \frac{T_t R_t}{L_A}$.

Each packet contains a header having H bits. Therefore, the overhead required to carry the headers is

$$O_{ps} = \frac{pnR_tH}{L_A} \text{ bits per second.}$$

The header for datagrams is larger than for virtual circuits.

4.3.1.3 Hybrid Switching Using SENET with Movable Boundary

Since the voice is carried by the circuit switched portion, the overhead will depend on the type of circuit switching. There is no overhead with traditional circuit switching. For circuit switching with TASI the overhead is as given in section 4.3.1.1.

4.3.1.4 Comparison

$$\frac{O_{TASI}}{O_{ps}} = \frac{2\,aL\,L_A}{T_tR_tH} = \frac{2L_A}{T_tR_t} \cdot a \cdot \frac{L}{H}$$

For the HF system, R_t cannot be greater than the maximum capacity of the links (which is 2400 bps). The value of L_A (which is 1048 bits) is dictated by the error rate on the links. The value of T_t (which is 0.6 seconds) has been obtained from Bullington and Fraser (Reference 2 p. 358).

Thus

$$\frac{O_{TASI}}{O_{ps}} = \frac{2 \times 1048}{0\cdot 6 \times 2400} \cdot a \cdot \frac{L}{H} = 1.46 \cdot a \cdot \frac{L}{H}$$

This result is plotted in Figure 3. The dotted line indicates that when $\frac{O_{TASI}}{O_{ps}}$ exceeds 1, then packet switching using either virtual circuits or datagrams is preferred over circuit switching with TASI, and when $\frac{O_{TASI}}{O_{ps}}$ is less than 1, circuit switching with TASI is preferred over packet switching using either virtual circuits or datagrams. In circuit switching with TASI, the primary contributors to overhead are the connect/disconnect messages. In packet switching, every voice packet carries a header so the overhead is dependent not only on the header length (H) but also on the number of these packets $\frac{R_t}{L_A}$.

4.3.2 Fractional Speech Loss

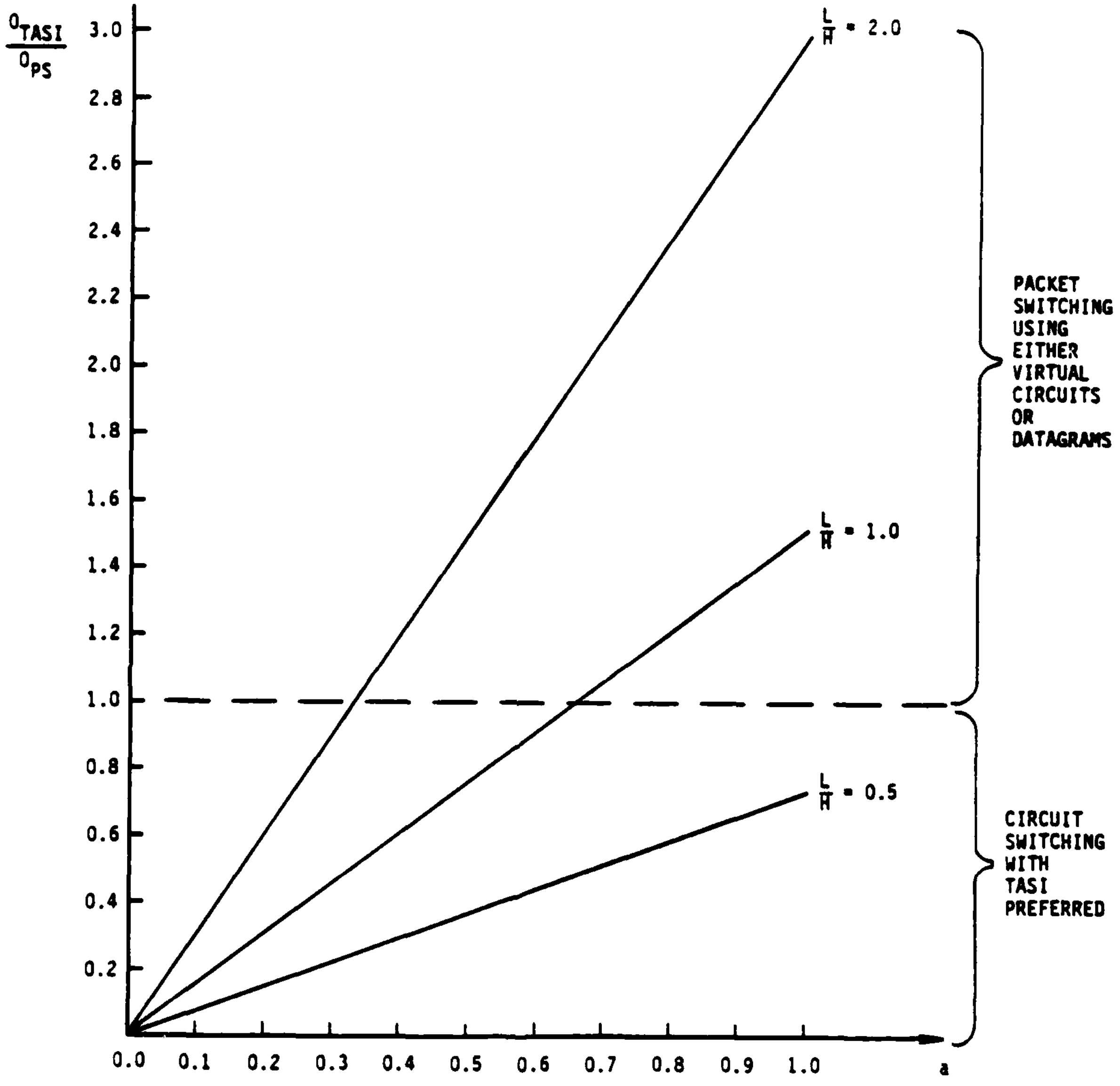
4.3.2.1 Circuit Switching with TASI

In this method, a number of speech sources share a smaller number of channels. Each source alternates at random intervals between talkspurts and silence. New talkspurts which begin at times when all channels are busy are "frozen out" and must wait on a first-come first-serve basis for an available channel. Once a channel is assigned to a particular talkspurt, that channel is held until the talkspurt ends. The occurrence of freeze-out typically causes the initial part of a talkspurt to be clipped. But a talkspurt can be lost entirely if the waiting time is longer than the talkspurt duration.

An important performance parameter of a TASI system is the fraction of speech lost due to freeze-outs, termed the cut-out fraction. The cut-out fraction does not depend on the details of either the talkspurt duration or silence period distribution but can be expressed in terms of the three system parameters:

1. n, the number of speech sources

2. c, the number of channels



PACKET LENGTH 1048 bits

TALKSPORT 600 ms

FIGURE 3: OVERHEAD COMPARISON BETWEEN TASI AND PACKET SWITCHING

3. p, the probability that a source is issuing a talkspurt at a random time.

A speech source refers to an input line to a TASI processor. Thus, p represents the probability that, at a random time, a particular line is busy and the talker on the line is issuing a talkspurt. Weinstein [5] has shown this cut-out fraction to be:

$$= \frac{1}{np} \sum_{k = C + 1}^{n} (k - c) \binom{n}{k} p^{k} (1 - p)^{n - k} .$$

This formula assumes that the speech sources are independent and is applicable at a single TASI processor, but does not hold for TASI processors in tandem.

4.3.2.2 Packet Switching Using Either Virtual Circuits or Datagrams

Consider a system in which n speech sources (each switching between talkspurt and silence) deliver packets to a single packet switching node and share, with equal priority, a single output line. The speech sources each transmit packets at the same constant rate (one packet every T_p seconds) while in talkspurt and none during silence. All packets are assumed to contain an equal number of bits.

The n speech sources can generate at most n packets every T_p seconds. The number of packets generated will range from 0 to n. The output line can transmit C packets every T_p seconds. The speech sources, occasionally generate more than C packets every T_p seconds. These excess packets cause a buildup of the packets at the switch, resulting in speech delay. A strategy to reduce this delay involves discarding packets in the switch queue after time T_d. If k packets are produced in time T_p, then $k \frac{Td}{T_p}$ packets will be produced in time T_d. Similarly, if C packets are transmitted in time T_p, then $C \frac{T_d}{T_p}$ packets will be transmitted in time T_d. Then, if $K \frac{T_d}{T_p}$ is greater than $C \frac{T_d}{T_p}$ packets are produced in time T_d, $(K - C) \frac{T_d}{T_p}$ of these packets, chosen at random from among the $k \frac{Td}{T_p}$ active talkers, will be discarded. If $K \frac{T_d}{T_p}$ less than $C \frac{T_d}{T_p}$ packets are produced in time T_d, all will be transmitted. The fractional speech loss associated with this scheme provides an overbound to the packet loss incurred in a scheme where overflow packets, in one T_d interval, are retained for transmission in the next T_d interval. Weinstein [5] has obtained an expression for the fractional speech loss. This result in a slightly modified form is:

$$= \frac{1}{np} \sum_{k = C + 1}^{n} (k - c)\, b\left(K \frac{T_d}{T_p}\ , n \frac{T_d}{T_p}\ p \right)$$

$$= \frac{1}{np} \sum_{k = C + 1}^{n} (k - c) \binom{n \frac{T_d}{T_p}}{K \frac{T_d}{T_p}} p^{K \frac{T_d}{T_p}} (1 - p)^{(n - k) \frac{T_d}{T_p}}$$

This formula describes packet loss through a single packet switch and does not deal with packet switches in tandem. The formula is equally applicable to virtual circuits as well as datagrams as the derivation does not take into consideration either the virtual circuit or datagram characteristics.

4.3.2.3 Hybrid Switching Using SENET with Movable Boundary

There is no fractional speech loss if the circuit switched portion of hybrid switching is traditional. On the other hand, if the circuit switched portion of hybrid switching uses TASI, then the fractional speech loss incurred by the voice traffic is as given in section 4.3.2.1.

4.3.2.4 Comparison

The fractional speech loss in both the TASI and the packet switched system is the same. In the packet switched system, all talkers will incur randomly distributed packet loss when more than C talkers are active. In TASI, however, the $(C + 1)^{st}$ and subsequent talkspurts to become active will be frozen out until a circuit becomes free, but previously active talkers will not be disturbed. The fractional speech loss in the hybrid switching technique is dependent on the method of voice switching that is used. If voice is packet switched, than the fractional speech loss is as given for the packet switched system. If voice is circuit switched using TASI then the fractional speech loss is as given for the TASI system. If voice is circuit switched using the traditional methods, there is no fractional speech loss.

4.3.3 Precedence/Preemption

4.3.3.1 Circuit Switching with TASI

A call may be assigned a precedence level which is used to resolve the conflict between two calls attempting a simultaneous set up: the higher precedence call is set up before the lower precedence call. In circuit switching, precedence/preemption is implemented on a loss basis, that is, the preempted calls are lost. Preemption occurs whenever, with all capacity used, a higher precedence call requests capacity. The lower precedence call will get preempted (which, in this case, means lost). Of course the user may retry at a later time.

The precedence level of a call can be used to determine if the call can be placed initially, i.e., with the capacity all used up, can the call preempt another call of lower precedence. In addition, the precedence level can protect a call from being preempted once it has been established by calls of lower (or equal) precedence.

4.3.3.2 Packet Switching Using Either Virtual Circuits or Datagrams

In packet switching, precedence/preemption is implemented on a delay basis. High priority voice packets are transmitted before low priority voice packets. The low priority calls will suffer degradation but the calls will not be cut off. Also, in the virtual circuit mode, high priority call set ups occur before the low priority call set ups; this precedence is analogous to circuit switching. In fact, low priority calls may be blocked.

4.3.3.3 Hybrid Switching Using SENET with Movable Boundary

When hybrid switching network resource become temporarily restricted or saturated, considerations of precedence become very important. Hybrid switching traffic is managed on a loss or blocking basis for the circuit switched traffic and on a delay basis for the packet switched traffic. Thus, for example, higher precedence data traffic, that could cause preemption of lower precedence voice traffic in a circuit switching concept, could be delayed in a hybrid switching concept. Thus, networks which employ hybrid switching should be capable of a better precedence/preemption performance than circuit switching networks because they are able to tailor their precedence/preemption strategy to the nature of the voice and data traffic statistics.

An issue to be resolved is the reconciliation of the precedence levels between voice and data, namely for equal precedence levels which type of call should be favored - voice or data. Preemption of voice calls is more severe than data calls. A data call which is preempted may only result in data packets being delayed and queued for later transmission. A preempted voice call required the subscriber to perform another dial up. For these reasons, data calls are preempted instead of voice calls, in the situation described above.

In the case of voice calls another question arises due to the fact that the CCIS messages for handling the call are themselves handled as packet switched data. These messages could be assigned to precedence level of the voice call.

4.3.4 Voice Continuity

4.3.4.1 Circuit Switching with TASI

The traditional circuit switching network provides a direct synchronous path between communicating subscribers. Thus, it provides an excellent medium for voice continuous operation. However, voice continuity performance degrades with the use of TASI because of the various forms of clips experienced by voice calls. In general, voice continuity performance is inversely related to the transmission efficiency and can be easily controlled.

4.3.4.2 Packet Switching Using Either Virtual Circuits or Datagrams

The characteristic of packet switching that affects voice continuity is primarily delays incurred by packets in the network. These include:

1. Packet assembly delay.

2. Network delay.

3. Packet reassembly delay.

The packet assembly delay (e.g., assembling a 1000 bit packet from a 2.4 kbit/sec terminal imposes over 400 ms transmission time) implies the use of shorter packets [6]. The resulting excessive overhead is reduced in the virtual circuit case (due to the fixed, call-oriented routing scheme), by the use of abbreviated packet headers. Thus, at call set up certain addressing and control information is stored in the nodes along the fixed route for subsequent use by the packets following this route. Overhead and delays are further reduced by lack of node to node and end to end control. Another result of using the virtual circuit approach is that the fixed call-oriented routing ensures the elimination of some components of network delays present in adaptive routing (such as due to variable path lengths encountered by packets). Furthermore, packets should not arrive out of order over a virtual circuit. However, average end to end delays may now be longer because the benefits of having multiple paths, for the packets to reach the destination, no longer exist.

With datagrams, the packet headers are long because they need to carry both source and destination addresses as well as sequencing information. In addition, the packets are individually routed through the network. Thus, the overhead and network delays are high. Both the overhead and network delays are somewhat reduced by lack of node to node and end to end error control. The beneficial effect of having multiple paths, for the packets to reach the destination, are possible with the use of datagrams.

Using virtual circuits in packet switching results in the packets being delivered in sequence and with reduced delays. Using datagrams in packet switching results in the packets being delivered out of sequence and with high delays. In the latter case, the packets have to be reassembled in sequence before delivery to the subscriber. In either case, Barberis and Pazzaglia [7] have indicated that the voice packets will arrive at the receiver asynchronously. This is because the network will introduce some stochastic delays resulting in gaps between packets (in the same talkspurt) arriving at the receiver. A packet voice receiver will compensate for these delays so that the output is nearly synchronous. It does so by introducing additional delays into the receiver. These delays are introduced by using a buffer to act as a matching unit between the asynchronous variable arrival rate and the requirement of synchronism. The specific algorithm of Barberis and Pazzaglia is discussed in Section 5.

4.3.4.3 Hybrid Switching Using SENET with Movable Boundary

Hybrid switching is able to retain the excellent voice continuity performance of traditional circuit switching. The strictly synchronous switching of voice using the SENET concept without TASI should give the best performance of the hybrid switching concept as far as voice continuity is concerned. Voice continuity performance degrades with the use of TASI.

4.3.4.4 Comparison

Hybrid switching with traditional circuit switching provides the best performance. Traditional circuit switching has the next best performance followed by circuit switching with TASI. Packet switching with virtual circuits and packet switching using datagrams present the most difficulties in maintaining voice continuity. This is for example due to the different paths travelled by different voice packets resulting in the delivery of out of sequence voice packets, wide delay variation, and the large packet overhead.

4.3.6 Error Control

4.3.6.1 Circuit Switching with TASI

No error control occurs on the node to node voice channels. This could lead to network wide transmission inefficiencies merely due to one high error link in the network. If any error control is desired, it must be accomplished on an end to end basis after the voice circuit has been set up. This is because the end subscribers have the processing flexibility to implement error control, whereas once a path is set up, the intermediate nodes are passive.

The CCIS channel does require error control on a link basis. Messages on the channel contain bits for error detection/correction. Messages in error are retransmitted. Consider two tandem nodes in which all the channels between them have been assigned to voice calls. In addition, another voice call is in the suspended state whereby, because it is in its silence period, no voice channel is assigned to it. If this voice call now has a talkspurt, node 1 will reassign one of its idle voice channels to this call. It will send a CCIS message to node 2 indicating the new channel assignment. The speed of the CCIS channel should be fast enough so that this message reaches node 2 before the voice on the newly assigned channel. If node 2 fails to receive this message, then, unless protective steps are taken, the swapping voice call will be directed to the swapped voice call (at node 2). One such protective step would be for node 1 to hold off sending the talkspurt on the newly assigned voice channel until it has received an acknowledgement from node 2 for the CCIS message. This results in a clip because the voice is blocked at node 1 until receipt of an acknowledgement. One method of reducing this clip is to increase the channel speed high enough for node 2 to receive a correct CCIS message and return an acknowledgement to node 1 before a noticable clip occurs. In addition, the threshold value after which node 1 retransmits the CCIS message (for lack of receipt of an acknowledgement from node 2) can be reduced.

Forward error correction techniques could be applied to the signaling messages on the CCIS channel. This requires additonal processing and/or transmission capacity; the accuracy that can be achieved will depend on the complexity of the error-correcting codes that are used. The advantage of using forward error correction is a reduction in the number of retransmissions and improvement in throughput/delay on the CCIS channel. Additionally, a particular complex forward error correction techique could reduce the probability of error in the received signals to a level where there would be no need for acknowledgements. This could help reduce the clip mentioned in the previous paragraph.

4.3.6.2 Packet Switching Using Virtual Circuit

Although error detection is employed, typically no error correction occurs on the node to node virtual circuit. Voice packets in error may be discarded by the nodes. Thus, any error correction needs are accomplished on an end to end basis after the virtual circuit is set up.

Error detection is accomplished by the presence of check bits in the voice packets. Instead of checking the entire voice packet, only the header portion of the packet need be checked. Since the voice portion of the packet contains redundant information, the presence of a few error bits does not affect the quality of the received voice appreciably. This is a preferable approach to one where the entire voice packet is discarded if the check bits (checking the entire voice packet) detect a few voice bits in error. In the first case, the received voice may contain only a few error bits whereas in the second case the received voice may lack entire voice packets. However, it is necessary to detect error in the header portion of voice packets because they contain critical information such as connection ID;

this could result in packets being misrouted. If the check bits detect any errors in the header portion, the entire voice packet is discarded. The option of error detecting only the header portion of a voice packet instead of the entire packet adds complexity to the error detecting logic.

The control packets (such as call request, call confirmation, etc.) contain bits for error detection. These packets are retransmitted if an error is detected. Since the control packets are transmitted before and after a call, the only effects of retransmissions are delays in setting up and/or bringing down a virtual circuit.

Forward error correction techniques may be applied to the headers of the voice packets. The technique can result in fewer voice packets being discarded at the expense of additional complexity, cost and greater delay. Forward error correction techniques could also be applied to the control packets. This can result in fewer retransmissions and thus faster call set ups and disconnections again at the expense of additional complexity and cost.

4.3.6.3 Packet Switching Using Datagrams

No error correction occurs on the node to node transmissions of datagrams. Voice datagrams in error may be discarded. Thus, any error correction requirements are accomplished on an end to end basis. It is possible to improve upon the error control mechanism as described above. A datagram contains a header and a large information text. Instead of checking the entire datagram for error, it is possible to only check the header portion of the datagram for error. Thus, the presence of a few error bits in the voice portion of a datagram is more preferable to having the entire datagram discarded because the bit representation of voice contains redundancy; the extent to which this is true depends on the voice coding technique. The effect on the quality of the received voice is more beneficial under the former scheme than the later. It is however, important to discard the datagram if the header contains errors because it is possible for the datagram to be misrouted by the presence of error bits. The option of error detecting only the header portion of a datagram instead of the entire datagram does add to the complexity of the error detection logic.

Also, forward error correction techniques may be applied to the header portion of the datagrams. The technique can result in fewer datagrams being discarded at the expense of additional complexity, cost, and greater delay.

4.3.6.4 Hybrid Switching Using SENET with Movable Boundary

Hybrid switching in which the circuit switched part uses TASI (and where the CCIS messages are packet switched) will have the same error characteristics as circuit switching with TASI. However, potentially hybrid switching in which the circuit switched portion is traditional will be able to supplement traditional circuit switching's weakness in this area. This is because packet switching will enable the inclusion of the acknowledgement scheme, retransmission scheme and flow control scheme into the traditional circuit switching system.

4.3.6.5 Comparison

In the sections above the issues associated with each technique in high error environments are discussed. However, optimal resolution of these issues is an area for further study.

5. Presentation Level Protocols

5.1 Voice Digitization

In order for voice to take advantage of the new techniques it may need to be digitized. Digitized voice can intermittently use the same equipment and lines as data without requiring expensive analog devices and frequency division multiplexers. Furthermore, digitized voice, using digital repeaters and detectors, provides enhanced transmission quality, distance independence and relative immunity to noise. Digital encryption techniques can be used for highly secure voice transmission.

Various technologies exist for the digitization of voice at different rates and costs. However, traditional digitization schemes are not appropriate. For example, Pulse Code Modulation (PCM) schemes sample the voice waveform at regular intervals (approximately 8 kHz) and associate every sample with a number referring to its quantized amplitude (8 bits/sample). Thus, the Voice Digitization Rate (VDR) for this PCM scheme is 64 kbps. Thus, PCM requires a large channel capacity. Since such capacity is unavailable in NTS, the use of lower rate speech coding techniques. These fall into two major classes, called waveform coders and source coders (VO coders).

The waveform coder approach to speech compression involves techniques for essentially directly coding the speech waveform. Several algorithms have been developed which take advantage of the correlation properties of the speech waveform. The basic premise is that fewer bits are needed to code the derivative of the speech than the speech itself. Therefore, instead of coding individual speech samples as in PCM, the differences between adjacent samples are coded for transmission.

There is a direct correlation between complexity, cost, and transmission rate of speech coders. Source coders are analysis-synthesis systems based on models of the human speech production process. In contrast to waveform coders, parameters extracted from the speech waveform, rather than the coded waveform itself, are transmitted to the receiver in a frame-like fashion. The selection of a speech coding algorithm for a given application depends on bit rate constraints, voice quality requirements, acoustic noise environments, and size, weight, and power supply considerations. As a general rule, as bit rates decrease, the complexity of the encoding/decoding hardware goes up, voice quality is reduced, and the deleterious effects of noisy or distorted inputs on speech quality and intelligibility are magnified.

Aoyama et al [9] have mentioned that linear predictive coding (LPC) is attractive as a means of reducing voice bandwidth, but that voice quality degrades and the coders have a high cost. Adaptive delta modulation (ADM) and DPCM are also effective in reducing bandwidth, although detection of a speech energy burst is difficult compared to PCM. ADM or DPCM coders are adopted on multiple voice channel operation.

The NTS network operates at the low rate of 2400 bits per second. Source coders such as LPC are suitable for use in the NTS network. The low rate of 2400 bits per second implies that improvements in voice quality may be obtained at the expense of complex encoding/decoding hardware.

5.2 Multirate or Variable Rate Algorithms

In any voice network there will always be a finite probability of having more channel requests than instantaneous available channel capacity. The standard flow control mechanism for accommodating this situation is to block call requests during overload. However, Bially, McLaughlin, and Weinstein [8] have pointed out that advanced speech coding techniques offer several flow control alternatives in the form of multirate or variable rate algorithms. The Navy has done extensive work in the field of multiple rate processor (MRP); MRP hardware is presently available. The hardware is based on the DOD standard speech coding technique, LPC-10. [11]

The ability to automatically select a rate at dial up or to vary the rate during a conversation in a manner that is transparent to the user (other than detectable change in quality) provides an additional degree of freedom in the design of integrated networks. In a particular voice rate assignment strategy that has been studied in the context of a SENET scheme, voice bit rates are assigned to new callers based on the percentage of the maximum allowable voice bandwidth that is already accounted for by other voice users at the time calls are initiated. Since data traffic shares the available SENET bandwidth with voice, it may also be appropriate to base voice rate assignments on data queue conditions along the voice route. The ability to operate at a variety of voice rates as a function of traffic load enhances the voice handling capability of the system, independent of the effect on data traffic. For example, the assignment of lower bit rates to new voice users as the number of active calls increases results in reduced blocking probability for voice calls during busy periods.

A more dynamic approach to voice flow control can be evolved by allowing bit rates to change during actual conversation. When the bit rate of a conversation changes, the circuit bandwidth has to change accordingly. This implies substantial control at the switching nodes and voice terminals that can be very rapidly reconfigured to support a number of different bit rates and perhaps different voice digitization algorithms. But this approach requires a finite time to sense overloads at the switching nodes, communicate flow control actions to individual voice terminals, and finally, to effect algorithm changes at the terminals themselves. Thus, during transient periods, terminals may still be operating at high bit rates while network links are saturated. This problem is eased by constraining the rate changes to occur during silence intervals in conjunction with speech activity detection. The concept of dynamically variable voice rates is easier to visualize in an all-packet-switched network, where the circuit management problems translate to packet header protocols and instantaneous voice link overloads can be alleviated by appropriate queueing and buffering actions. Specifically, the switching node, on sensing overhead, will send high priority flow control packets to the individual voice terminals, to effect a reduction in their bit rates. During the finite time required for these packets to reach the voice terminals, the conversations continue at the high bit rates. The excess voice packets accumulate at the switching nodes in buffers. As the voice terminals reduce their bit rates, buffer overflow at the nodes is avoided and as the congestion works off, buffer usage reduces.

5.3 Voice Packetization/Reassembly Algorithm

The packetization of digital voice is a relatively simple operation, consisting of framing voice packets from the incoming voice bits. They contain an abbreviated header and voice bits enveloped by flags in the virtual circuit scheme and a full header with voice bits enveloped by flags in the datagram scheme. Also, bits for error detection are also included; these bits may protect the entire packet (excluding framing bits) or only the header. The voice packets are transmitted either when they are full or on detection of a pause in speech.

The reassembly of packets into digital voice is done by a packet voice receiver (PVR). This device must compensate for the stochastic delays experienced by packets in the network and delivering a nearly synchronous voice output that is intelligible to the human ear.

In order to achieve perfect voice reconstruction in a packet switched environment, the PVR needs complete timing information. The substance of the packet switching problem is twofold:

- Reconstruct the speaker silence period between talkspurts at the receiver; this is done by appropriately delaying the first packet in a talkspurt.

- Maintaining voice continuity by providing prompt delivery of subsequent packets (without gaps).

One approach to solving this problem has been developed by Barberis and Pazzaglia [7] is to artificially delay the first packet; this is described below. Let

T = threshold parameter

t_o = network transit time.

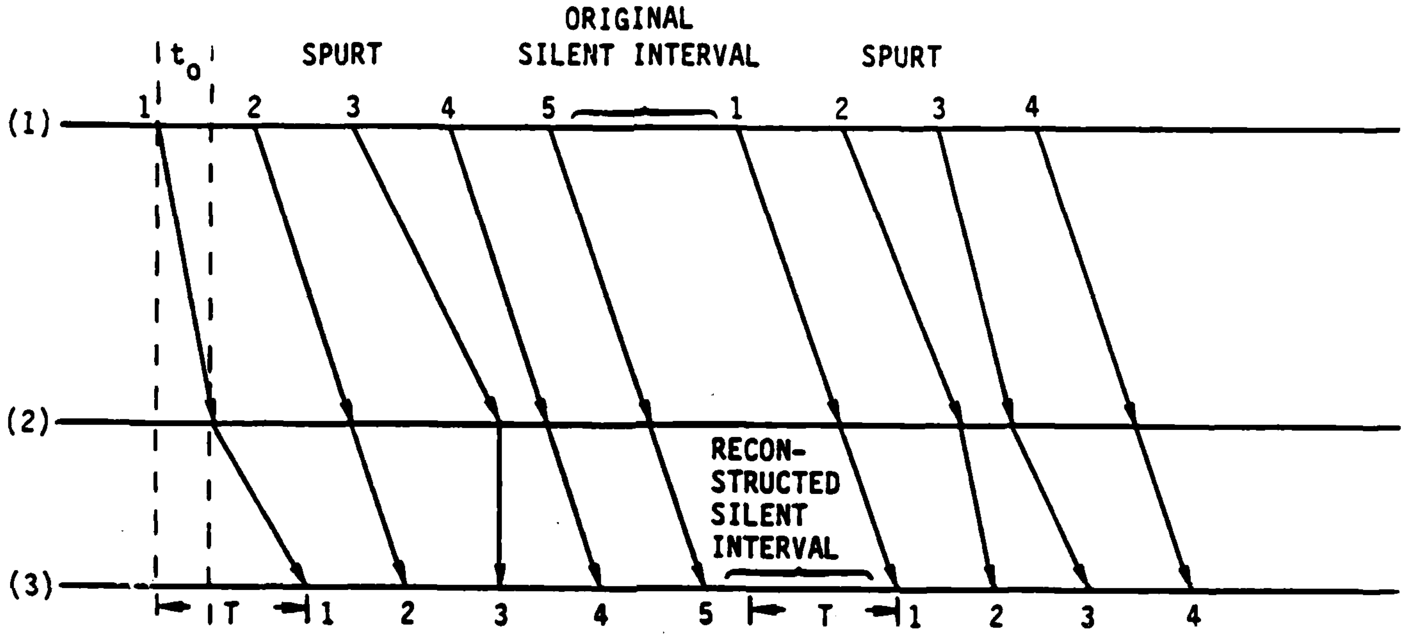
The PVR would delay every first packet of a talkspurt, received from the network, by an amount $T-t_o$, if the network transit time, t_o, of this first packet, is below a threshold control parameter, T. The parameter T changes according to the estimated network stochastic parameters. If, on the other hand, the network transit time is greater then the threshold control parameter, the device does not delay the first packet of the talkspurt. This may be summarized by saying the total delay, t, effected by transit and the PVR would be

$t = T$ *if* t_o *is less than* T

$t = t_o$ if t_o is greater than T.

Subsequent packets in a talkspurt are sent to the A/D converter immediately after completion of the previous packet. Thus, it is the responsibility of the network to guarantee the arrival of these packets by the time the PVR is ready to send the packet to the A/D converter. The behavior of this algorithm is depicted in Figure 4. It depicts generation of the packet (1), arrival of the packet at the receiver (2), and forwarding of the packet to the A/D converter (3). For the first talkspurt depicted, the first packet is delayed by the threshold value T because the transit time was less than T. All subsequent packets in the first talkspurt are immediately forwarded to the A/D converter. Similarly, for the second talkspurt t_o is less than T, and a combined delay of T is effected. This mechanism results in a good cancellation of gaps introducted by the stochastic network between packets of the same talkspurt.

However, there exists a lack of synchronism between the sender and the PVR because the clocks in the sender and receiver may not use the exact same time reference. Hence, it



(1) PACKET BIRTHS

(2) PACKET ARRIVALS AT THE P.V.R.

(3) PACKET TOWARDS THE D/A DEVICE

FIGURE 4: TIME DIAGRAM FOR PACKET REASSEMBLY

is not possible to exactly determine t_o. One solution to this problem is to introduce a global clock. But this is an impractical solution. An alternate solution, which will be described here, is to synchronize the clocks at the sender and the PVR, with the smallest delay measured for the packets in a talkspurt. This is an adaptive procedure in the sense that a measured delay which is smaller than any previously measured delay will cause a synchronization. After the transit time is estimated, the algorithm described above is applied to determine the quanta for delaying the first packet of a talkspurt.

The synchronization algorithm that reduces the gap between the sender and the PVR clocks to zero involves empirically estimating the time delays. For each first packet of every talkspurt, the birth time stamp is coded in the header, according to the time base of the sender's clock. This value will be read by the PVR with another time base and thus an error in the transit delay estimate, t_o, is introduced; if the gap is removed, the PVR knows the exact transit time.

The time base of the sender's clock starts when the first packet of the call is sent. The time stamp, t_s, coded in the header of the first packet of the first talkspurt will be set at zero. When this packet joins the PVR, its clock is switched on, taking into account the deterministic delay (transmission and switching times) experienced in the network path by the packet; this means that the stochastic delay is assumed to be zero. In this way, the gap between the two clocks equals the random delay of the first packet of the call. The time stamps of the following packets, of the first talkspurt are read by the PVR according to the time base of its clock. In this way, whenever the random delay of a packet is lower than the random delay of a previous one, the time clock of the receiver assumes a value lower than the possible one, and so, a clock correction is made. In this way, the receiver clock always runs behind the sender clock.

For each subsequent packet the receiver will make a new estimate for its clock. Let t_s be the time stamp of received packet, D be the deterministic delays, and t_r be the time at which the receiver received the packet. Then

if $t_r < t_s + D$, the receive clock is reset; $t_r = t_s + D$

and if $t_r > t_s + D$, no change is made,

or, equivalently, the new estimate of time in the receiver is

$$t_r = \text{maximum} \left\{ t_r, t_s + D \right\}.$$

After the clock is adjusted, the algorithm for delaying the first packet described above is applied. The transit time is estimated as

$$t_o = t_r - t_s$$

using the latest value of t_r and the timestamp t_s in the packet.

6. CONCLUSIONS

In this volume, we have highlighted the technical issues associated with voice/data integration in NTS. The key issues identified are associated with switching and associated signaling techniques and presentation layer protocol, and include

- Relative performance efficiency (in terms of in-band signaling versus out-of-band signaling, this involves both the overhead bits and the handling of two types of traffic (information and control) with different characteristics.

- The performance of these signaling schemes in a high error rate, dynamic environment; the use of forward error correction and separate header error detection to enhance performance.

- The complexities of co-ordinating the out-of-band signals (in the presence of errors) with the in-band transmission of information, this will involve both hardware and software co-ordination.

- The performance of reassembly algorithms for packet voice to provide intelligible conversation.

- The use of variable rate coding as flow control procedures.

In summary, accomodation of the diverse traffic types in future NTS networks is a major problem. Hence, it is critical for future NTS systems to optimize the use of voice/data integration techniques in order to most efficiently utilize the available channel capacity. However, the technical issues enumerated above must be more thorougly studied. In particular the network (as opposed to link) performance of these issues must be investigated; for example the signaling and information channel network topology could be different.

References

1. Brady, P.T., "A statistical analysis of on-off patterns in 16 conversations," Bell System Technical Journal, vol. 47, January 1968.

2. Bullington, K., and Fraser, J., "Engineering aspects of TASI," Bell System Technical Journal, pp. 353-364, March 1959.

3. Gaver, D.P., and Lehoczky, J.P., "Channels that cooperatively service a data stream and voice messages," IEEE Trans. Commun., vol. COM-30, pp. 1153-1162, May 1982.

4. Fischer, M., "Delay analysis of TASI with random fluctuations in the number of voice calls," IEEE Trans. Commun., vol. COM-28, pp.1883-1889, November 1980.

5. Weinstein, C., "Fractional speech loss and talker activity model for TASI and for packet-switched speech," IEEE Trans. Commun., vol. COM-26, pp. 1253-1257, August 1978.

6. Gruber, J.G., "Delay related issues in integrated voice and data networks," IEEE Trans. Commun., vol. COM-29, pp. 786-800, June 1981.

7. Barberis, G., and Pazzaglia, D., "Analysis and optimal design of a packet-voice receiver," IEEE Trans. Commun., vol. COM-28, pp. 217-227, February 1980.

8. Bially, T., McLaughlin, A., and Weinstein, C., "Voice communication in integrated digital voice and data networks," IEEE Trans. Commun., vol. COM-28, pp. 1478-1490, September 1980.

9. Aoyama et al, "Packetized service integration network for dedicated voice/data subscribers," IEEE Trans. Commun., vol. COM-29, pp. 1595-1601, November 1981.

10. Ross, M. and Mowafi, O., "Performance analysis of hvbrid switching concepts for integrated voice/data communications," IEEE Trans. Commun., vol. COM-30, pp. 1073-1087, May 1982.

11. Department of Navy Multiple Rate Processor (MRP) program report.

END

FILMED